THE USENIX ASSOCIATION NEWSLETTER

;login:

AUGUST 1994 • VOLUME 19 • NUMBER 4

DO NOT REMOVE
FROM COAST LAB

CONTENTS

FROM THE EDITOR

Technology Marches On 3

USENIX NEWS

USENIX Resource Center on WWW 3
Hot Off the Press – 4.4 BSD Manuals & CD-ROM .. 4
UNIX Applications Development Symposium Report *by Stephen Walli* 5
Summer '94 Conference Reports *by Pace Willison and Jerry Peek* 8
Board Meeting Summary 14
Letter to the Editor *by John Lockwood* 15
Executive Office Staff Addition 15
Community News 15
Know Your Board and Staff 16
;login: 50 and 100 Years Ago *by Barry Shein* 45

SAGE NEWS

Letter from the SAGE President 17
by Elizabeth D. Zwicky
From the Editor *by Bryan McDonald* 17
Highlights from the June SAGE Board 18
by Paul Evans
Call for Nominees for Election to SAGE Board of Directors *by Pat Wilson* 18
SAGE-AU Code of Ethical Conduct 19
by Kate Lance
Please Unsubscribe Me from the List... 20
by Bryan McDonald
System Administration Tools Your Vendor Never Told You About: The Beeper 21
by Elizabeth D. Zwicky
Who Are My Peers 22
by Elizabeth D. Zwicky

FEATURES

Celebrating Women in Computing 23
by Vicki E. Jones
Beneath Who Knows How Many Freudian and Calvinistic Strata *by Jean Pooquet* 25
Optimizing Your Shell Scripts 25
by Scott Hazen Mueller
Conquering Corporate Computing With Message-Oriented Middleware 28
by Tim Daneliuk
What's New 32

STANDARDS REPORTS

An Update on Standards Relevant to USENIX Members *by Nick Stoughton* 37

BOOK REVIEWS

The Bookworm *by Peter H. Salus* 41
Ethical Conflicts in Information and Computer Science, Technology, and Business – Reviewed by Rob Kolstad 42
Internetworking with TCP/IP – Reviewed by George Neville-Neil 43
Connecting to the Internet – Reviewed by Adam S. Moskowitz 44
!%@::A Directory of Electronic Mail Addressing Networks (4th Edition) 45

ANNOUNCEMENTS & CALLS

LISA 1994 Program 46
Calls for Papers:
- COOTS 49
- Symposium on Mobile and Location-Independent Computing – 1995 51
- Workshop on Mobile Computing Systems and Applications – 1994 53
- GURU's 1994 Conference 54

ETC...

Book Publishers Discount Order Forms 55
Proceedings Order Form 61
Subscription Order Form 63
Local User Groups 64
Calendar of Events 66

USENIX The UNIX and Advanced Computing Systems Professional and Technical Association

;*login:* is the official newsletter of the USENIX Association.

•;*login:* (ISSN 1044-6397) Volume 19, Number 4 (August 1994) is published bimonthly by the USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

• $24 of each member's annual dues is for an annual subscription to *;login:*. Subscriptions for nonmembers are $50 per year.

• Second-class postage paid at Berkeley, CA and additional offices.

• POSTMASTER: Send address changes to *;login:*, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

EDITORIAL STAFF

• Rob Kolstad, Editor
<*kolstad@usenix.org*>

• Bryan McDonald, SAGE Editor
<*bigmac@erg.sri.com*>

• Nick Stoughton, Standards Report Editor
<*nick@usenix.org*>

• Carolyn S. Carr, Managing Editor & Typesetter
<*carolyn@usenix.org*>

• Lilia Scott, Proofreader

MEMBERSHIP AND PUBLICATIONS

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
Telephone: 510/528-8649
FAX: 510/548-5738
Email: <*office@usenix.org*>

CONTRIBUTIONS SOLICITED

You are encouraged to contribute articles, book reviews, and announcements to *;login:*. Send them via email to <*login@usenix.org*> or through the postal system to the Association office. Send SAGE material to <*bigmac@erg.sri.com*>. The Association reserves the right to edit submitted material. Any reproduction of this newsletter in its entirety or in part requires the permission of the Association and the author(s).

Copyright © 1994 USENIX Association. USENIX is a registered trademark of the USENIX Association. UNIX is a registered trademark licensed exclusively to X/Open Company, Ltd. The Association acknowledges all other trade references made herein.

;login: is produced with Framemaker 4.0 software provided by Frame Technology and displayed on an NCD X Terminal, donated by Network Computing Devices. The system is served by a Sun SPARCsystem 10 server. Output is sent to a 600 dpi QMS 860 laser printer, donated by Quality Micro Systems. The newsletter is printed on recycled paper.

UPCOMING SYMPOSIA AND CONFERENCES

SEPTEMBER 19–23, 1994

8TH USENIX SYSTEMS ADMINISTRATION CONFERENCE

Co-sponsored by SAGE, the System Administrators Guild

San Diego, California

Program Chair: Dinah McNutt, *Zilker Internet Park, Inc.*

LISA VIII

OCTOBER 26–28, 1994

USENIX SYMPOSIUM ON VERY HIGH LEVEL LANGUAGES (VHLL)

Santa Fe, New Mexico

Program Chair: Tom Christiansen, *Consultant*

NOVEMBER 14–18, 1994

USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION (OSDI)

Co-sponsored by ACM SIGOPS and IEEE TCOS

acm. IEEE COMPUTER SOCIETY

Monterey, California

Program Chair: Jay Lepreau, *University of Utah*

DECEMBER 8–9, 1994

WORKSHOP ON MOBILE COMPUTING SYSTEMS AND APPLICATIONS

Sponsored by IEEE TCOS in cooperation with ACM SIGOPS and USENIX

IEEE COMPUTER SOCIETY acm. USENIX

Santa Cruz, California

General Chair: Darrell Long, *University of California, Santa Cruz*

Program Chair: M. Satyanarayanan, *Carnegie Mellon University*

JANUARY 16–20, 1995

USENIX WINTER 1995 TECHNICAL CONFERENCE

The only "broad-topic" USENIX Conference in 1995!

New Orleans, Louisiana

Program Chair: Peter Honeyman, *CITI, University of Michigan*

APRIL 10–11, 1995

2ND USENIX SYMPOSIUM ON MOBILE & LOCATION-INDEPENDENT COMPUTING

Ann Arbor, Michigan

Program Chair: Jim Rees, *CITI, University of Michigan*

JUNE 26–29, 1995

USENIX CONFERENCE ON OBJECT ORIENTED TECHNOLOGIES (COOTS)

Monterey, California

Program Chair: Vince Russo, *Purdue University*

Tutorial Program Chair: Doug Lea, *SUNY Oswego*

SEPTEMBER 18–22, 1995

9TH USENIX SYSTEMS ADMINISTRATION CONFERENCE (LISA IX)

Co-sponsored by SAGE, the System Administrators Guild

Monterey, California

Program Chair: Tina Darmohray, *Lawrence Livermore National Laboratory*, & Paul Evans, *Synopses, Inc.*

LISA

TO RECEIVE FULL INFORMATION

Please contact: USENIX Conference Office
22672 Lambert St., Suite 613, Lake Forest, CA USA 92630
1 (714) 588-8649; FAX: 1 (714) 588-9706 e-mail: conference@usenix.org

FROM THE EDITOR

Technology Marches On

I frequently write about how I'm just amazed by recent technology developments. Today's column comes to you from the microcomputer lab at the University of Wisconsin, Parkside, near Kenosha and Racine. The entire lab, of course, has telnet and ftp access to the Internet. I can telnet to work or home and continue my work from this great distance.

I was marveling about this and other developments (high quality audio CDs, house construction techniques, and high tech automobiles, to name a few) to my friend Donn. I asserted that things really were getting better and life was better each decade in each and every way.

"Yeah," he mused, "Most things are better. I miss one thing about the good old days, though. You just can't park your bike any more without locking it up real tight. Likewise, you can't leave things around campus because people just pick them up. It's like a little game, 'No owner attached, it must be mine.' It didn't used to be like that."

Of course, he's right. There are so many good things about our fine country. I wonder how we went wrong and ended up in the current state of affairs vis à vis loose property. I do not personally know of other locales on our fine planet that are better–my friend Greg notes that his wallet was stolen in Iceland.

What a drag. Maybe that's something we should work on for the 21st century.

RK

USENIX Resource Center on the World Wide Web

The USENIX Association Resource Center went online June 1, so if you have a World Wide Web (WWW) browser (such as Mosaic or Lynx), you can access it at any time with the URL: `http://www.usenix.org`

The Resource Center will initially have information on current activities of SAGE and USENIX; how to join; a calendar; conference announcements, programs and registration forms; samples of articles from past issues of the newsletter; ordering information for publications; and access to the USENIX Online Library and Index. In the future we hope to make full papers and abstracts from recent conferences available to members.

If you do not have access the World Wide Web, you can also obtain information about USENIX from our automatic mailserver. To get the catalog listing of what is available, please send mail to: `info@usenix.org`.

The body of your mail message should say: `send catalog` and the catalog will be returned to you.

USENIX ASSOCIATION BOARD OF DIRECTORS

If a member wishes to communicate directly with the USENIX Board of Directors, he or she may do so by writing to <*board@usenix.org*>.

President:
- Stephen C. Johnson <*scj@usenix.org*>

Vice President:
- Eric Allman <eric@usenix.org>

Secretary:
- Lori Grob <grob@usenix.org>

Treasurer:
- Rick Adams <*rick@usenix.org*>

Directors:
- Tom Christiansen <*tchrist@usenix.org*>
- Dan Geer <*geer@usenix.org*>
- Andrew Hume <*andrew@usenix.org*>
- Greg Rose <*ggr@usenix.org*>

Executive Director
- Ellie Young <*ellie@usenix.org*>

CONFERENCE & SYMPOSIA

- Judith F. DesHarnais
  Conference Coordinator
  USENIX Conference Offic
  22672 Lambert Street, Suite 613
  Lake Forest, CA 92630
  Telephone: 714/588-8649
  FAX: 714/588-9706
  Email: <*conference@usenix.org*>

- Zanna Knight
  Vendor Display Coordinator
  Email: <*zanna@usenix.org*>

TUTORIALS

- Daniel V. Klein
  Tutorial Coordinator
  Telephone: 412/421-2332
  Email: <*dvk@usenix.org*>

AUTOMATIC INFORMATION SERVER

To receive information electronically about upcoming USENIX symposia & conferences finger <*info@usenix.org*> and you will be directed to the catalog which outlines all available information about USENIX services.

SUPPORTING MEMBERS

ANDATACO
ASANTÉ Technologies, Inc.
Frame Technology Corporation
Matsushita Electrical Industrial Co., Ltd.
Network Computing Devices, Inc.
OTA Limited Partnership
Quality Micro Systems, Inc.
Tandem Computers, Inc.
UUNET Technologies, Inc.

The closing dates for the next two issues of *;login:* are August 15 and October 14, 1994.

USENIX MEMBER BENEFITS

As a member of the USENIX Association, you receive the following benefits:

- Free subscription to *;login:*, technical features, system administration tips and techniques, international calendar of events, SAGE News, media reviews, Snitch Reports from the USENIX representative and others on various ANSI, IEEE, and ISO standards efforts, and much more.
- Free subscription to *Computing Systems*, the refereed technical quarterly published with The MIT Press.
- Discounts on registration fees for the large, multi-topic Winter and Summer technical conferences, the System Administration conference (LISA), and the various single-topic symposia addressing topics such as object technology, security, operating systems, high-speed networking, and mobile computing – as many as eight technical meetings every year.
- Special subscription rates to the periodicals *Open Computing* (the new name for *UNIX World*) and *The LINUX Journal*.
- Discounts on proceedings from USENIX conferences and symposia and other technical publications. Discounts on the USENIX Association book series published by The MIT Press. Now available: *The Evolution of C++: Language Design in the Marketplace of Ideas*, edited by Jim Waldo of Sun Microsystems Laboratories.
- Discounts on the five volume set of 4.4 BSD manuals plus CD published by O'Reilly & Associates and USENIX.
- Savings (10-20%) on selected publications from McGraw-Hill, The MIT Press, Prentice Hall, John Wiley & Sons, O'Reilly and Associates, and UniForum.
- Right to join SAGE, the System Administrators Guild

To become a member or receive information regarding your membership status or benefits, please contact <*office@usenix.org*> or telephone (510)528-8649

Hot Off the Press – 4.4 BSD Manuals and CD–ROM

The Computer Systems Research Group of the University of California at Berkeley, in cooperation with the USENIX Association and O'Reilly & Associates, Inc, has now published the final, definitive release of the Berkeley version of UNIX, which has been the basis for many commercial UNIX variants. The source code for the system is available on CD-ROM and is accompanied by a five-volume documentation set. USENIX members who wish to purchase these will receive a special discount. See prices on the next page.

4.4BSD-Lite CD-ROM Companion

With this release of 4.4BSD, one need no longer work at a university or UNIX system development house to have access to UNIX source. The University has prepared a "Lite" release of the operating system, from which the source code for a small number of utilities and files, including a few from the operating system, has been removed so that the system can be freely distributed. *(Editor's note: It is not bootable.)*

The source code included on the 4.4BSD-Lite CD-ROM Companion will provide invaluable information on the design of any modern UNIX or UNIX-like system, and the source code for the utilities and support libraries will greatly enhance any programmer's toolkit.

In addition to source code, the CD includes the manual pages, other documentation, and research papers from the University of California, Berkeley's 4.4BSD-Lite distribution.

4.4BSD Five-Volume Doc Set

A five-volume doc set is available from O'Reilly directly with or without CD-ROM. The doc set includes:

4.4BSD System Manager's Manual (SMM)

This volume includes man pages for system administration commands (section 8 of the online reference manual), plus supplementary documents useful to system administrators. Includes installation instructions for the 4.4BSD release (which are also included with the 4.4BSD-lite CD-ROM Companion), plus papers on many system administration utilities and tasks, many of which are the definitive works on the topic.

4.4BSD User's Reference Manual (URM)

This volume collects the man pages for the more than 284 user programs that make up the user portion of the 4.4BSD UNIX release (section 1 of the traditional UNIX Reference Manual), plus additional pages for games (section 6 of the traditional manual) and miscellaneous information (section 7, which includes mostly information on troff macro packages and other text processing information). URM is useful even if you don't use the BSD version of UNIX. Many of the commands are compatible with the POSIX standard and thus identical to those in other UNIX versions. Because 4.4BSD includes many freeware programs that are available but not documented in other versions of UNIX, you may find yourself turning first to this book to look up options for programs like perl, GNU emacs, compress, patch, g++, or kerberos.

4.4BSD User's Supplementary Documents (USD)

A collection of papers relating to miscellaneous "user" tools, principally text editors and document processors. Some remain the only tutorial documentation for many UNIX programs.

4.4BSD Programmer's Reference Manual (PRM)

The PRM collects the man pages for the various UNIX programming libraries, plus related information for programmers. It consists of sections 2 through 5 of the online man page collection: (2) UNIX System Calls; (3) C Library Subroutines; (4) Special Files (device interfaces); and, (5) File Formats. Useful not only for 4.4BSD systems but for many related UNIX implementations, including SunOS, BSDI, and Linux.

4.4 BSD Programmer's Supplementary Documents (PSD)

This volume contains useful papers on such tools as the gdb hand adb debuggers, source code control systems RCS and SCCS, lex and yacc, and the m4 macro processor. Also included are such historical documents as Dennis Ritchie and Ken Thompson's classic first paper about UNIX and Kernighan and Ritchie's classic introduction to UNIX programming. On a more up to date note, this book also includes a two part tutorial on interprocess communication (IPC) under 4.4BSD UNIX.

USENIX member discounted prices are:

URM	$24.00
SMM	$24.00
PRM	$24.00
USM	$24.00
PSD	$24.00
CD-ROM	$36.00
5 Volume Set + CD-ROM	$150.00

To place an order send email to *order@ora.com* or call 800/998-9938.

UNIX Applications Development Symposium Report

by Stephen Walli
<stephe@mks.com>

The USENIX UNIX Applications Development Symposium was held in Toronto, Canada, April 25-28. This was the event that was postponed from previous year due to problems pulling papers. The program presented in Toronto was excellent, however, so it was worth the wait. I believe this was a long overdue symposium. The program was directed at problems, large and small, with developing applications in a UNIX world. I've been involved with USENIX since Spring 1991, and I have continually heard concerns expressed that there exists a large group of people within the USENIX fold that aren't really addressed at the main technical symposia. They develop applications for a living, either in-house or shrink-wrapped for distribution. UniForum isn't technical enough for them, yet papers on the latest file systems or micro-kernels just don't help them get their job done. At the same time, no one wants to change the research focus of the main technical conferences.

These applications developers are also the likeliest first casualties of the recession with respect to attending USENIX conferences, as conferences are "fun" and often don't look enough like training to IS management. The Toronto event reflected this. Most of the eighty or so registrants were Canadian from the Toronto area, i.e., no travel allowances required.

Regardless of the small attendance of this first time event, I believe it will provide the foundation for better things to come. The small size certainly allowed attendees to participate easily in what are better described as discussion periods rather than question periods following papers.

An excellent group of papers was presented. The proceedings are well worth the price, and are available from the USENIX office. Without further ado, here is a brief summary of the papers presented in each session.

Robert Adams (Dux Software) kicked the conference off with a discussion of how fast technology has matured in the past few generations, relating it back to his own colorful family history in California. He closed with a video tape of the developer of the SimCity port discussing the graphical object technology used to port SimCity.

Wrapping DCE/OSF Client/Server Applications.
Israel Gold, IBM Israel Science and Technology.

Israel Gold described new work that makes it easier to build DCE applications. DCE (Distributed Computer Environment) is an OSF sponsored technology used to develop client server applications. Developing DCE applications is a non-trivial task, as it requires a detailed understanding of all aspects of the DCE model. Gold's paper describes a small language (APF) and a tool (gluegen) used to describe the interfaces, registration and bind mechanisms, etc., and then build the custom pieces of the application into the DCE world.

This work has the distinct feel of something like Tcl/Tk, where a small language and a widget set was used to allow developers to quickly build graphic front-ends for applications. Here it is the DCE API, rather than an X toolkit like Xt or Motif, that is being replaced with a more appropriate programming language at a different level entirely.

Dagger: A Tool to Generate Program Graphs. Yih-Farn (Robin) Chen, AT&T Bell Laboratories

Dagger is another tool in a growing body of work from Bell Labs based on the C Information Abstraction system (CIA). The system is used for static code analysis, and papers have been presented at USENIX conferences on various aspects of the work over the last few years. Dagger takes the output from queries on a program database, and generates program graphs of various forms and complexities on such things as header file hierarchies, type inheritance, and module bindings.

This paper presented the graph generation process, and looked at a number of ways to simplify complex graphs. Examples were drawn from large bodies of real-world code, (an X11 release being everyone's favorite). Pertinent timing information was included, allowing every one to see that a lot of information can be presented in very little time.

Developing Applications with a UIMS. Dan Klein, Lonewolf Systems.

Klein presented a User Interface Management System (UIMS) to partition an application into three components or domains; (i) the application domain, (ii) the display technology domain, and (iii) the user interface domain. Using this approach makes the individual parts of the application more maintainable and more available for reuse in new situations. The paper presented the general features of the *Alpha* UIMS, and how to develop an application using such a system.

SPP-Low Tech, Practical, UNIX Software Portability. John Sellens, University of Waterloo.

SPP is a simple collection of library routines, C language include files, and a few utility programs to deal with some of the simpler problems of porting software amongst UNIX variants. The system has been used at the University of Waterloo for a number of projects, but the author admits that this system should probably only be used for small scale problems.

This was my biggest concern with the paper: I don't believe it will scale well to larger projects, which leaves the user with either a mixed portability model, one for large projects and another for small projects, or having to extend the existing model, which would be difficult. There have been a number of excellent papers on source code portability approaches presented at USENIX events over the past few years. The SPP paper provides an interesting starting point to think about portability models, however, and bears reading.

Creating a Configurable Compiler Driver for System V Release 4. John F. Dooley, Motorola Computer Group.

A configurable compiler driver (CCD) for C has been developed for the System V Release 4 operating system shipped by Motorola. The CCD is table driven, and the user can alter the personality of the compiler by setting an environment variable. This gives the user the benefit of consistency at the command line, and hence in makefiles, while allowing different compilers (such as gcc or CI5) to be used depending on requirements at the time. This method also serves cross-compilations for other hardware, etc. CCD allows a user to add configurations for new C compilers to suit their own needs.

Jam-Make(1) Redux. Christopher Seiwald, Ingres Corporation.

Jam was developed to address many of the problems from which *make(1)* suffers. Before this paper is dispatched as yet-another-make (yam?) it should be noted that the author has done homework. (The bibliography includes most *make* papers from the past decade, with the exception of Mahler's shape papers, and Tilbrook's qef work.)

While Jam still uses time-stamps to determine what needs to be built, it goes along way to simplify the construction expression, and points out ways it would approach some of the other problems, such as caching header-file dependency information to improve performance.

Jam is being used to build Ingres in its production environment, and all the source is available in Volume 27, of comp.source.unix.

Writing, Supporting, and Evaluating Tripwire: A Publicly Available Security Tool. Gene H. Kim, Purdue University.

Tripwire is a systems administration tool used to check system integrity in UNIX environments, and is distributed freely on the Internet. This paper discussed how the "product," which began as an independent study program for an undergraduate, was shaped by very determined "customers" using it in unexpected ways. Distribution, support, and enhancements are discussed, along with the tools to manage Tripwire.

This paper is an excellent how-to (and mistakes learned) discussion of the informal release of Tripwire and customers' expectations for its support. It should be essential reading for anyone planning to distribute their code on the net.

Design, Distribution and Management of Object-Oriented Software. Dinesh Kulkarni, University of Notre Dame.

Object-oriented software ICs and software re-use have not lived up to the promise of a few years ago, and remain largely a myth. This paper presents research underway to attempt to solve part of problem with a three tiered attack:

- a flexible framework that allows extension to software components without re-compilation,
- partially resolved loadable subclasses which can be distributed as re-usable units,
- programmer guidelines for creating software that can be fine-tuned at run-time.

The work presented in the paper lays a foundation for future explorations in easing the problems of distributing re-usable modules of "object-oriented" software.

I apologize to those who presented papers in the "New Approaches for X" session. That was the one part of the conference I could not attend. I feel it was too bad there were no Tcl/Tk papers, as recent discussions at the Summer Conference in Boston seem to indicate that more people are starting to use it in real applications. Maybe next time.

One paper caught my eye in the proceedings: "The Xt Intrinsics as a General Purpose Application Development Platform, or a User Interface Toolkit With Optional Users," by Jordan Hayes and Charles Ocheret. The premise is that Xt provides a lot of useful facilities for developing distributed (soft) real-time applications that may not have a GUI, and its as portable as X11.

Bridging the Technology Generation Gap: Upgrading a Network Management Application to a New Technology Base. Jay S. Lark, Teknekron Communications Systems.

NMCS is a network management system for T-1 switches, which has been recently upgraded to an entirely new technology base, such that it can be more flexible to the changing demands of its customers. In developing and releasing NMCS II, the author's company discovered what it has called the "technology generation gap," the subjective end-user comparison of the current state-of-the-art, all-singing, all-dancing version of the product to the original low-tech product. The customers were not as enthused as one might initially imagine. This difference between the formal acceptance test and the customers' subjective measures of acceptability was discussed, as well as how Teknekron worked through these areas of difficulty.

Porting and Maintaining with X and Motif: A Retrospective View. Paul Davey, User Interface Technologies Ltd.

Davey presented an excellent case history of the problems encountered and lessons learned in managing the software base of IXI Ltd.'s X.desktop product. By 1992, X.desktop's success had the team supporting three major versions across 30 platforms. The case history covered aspects of developing and supporting multi-platform code, tool support for managing the construction and configuration of the product, and the development of problem and bug tracking systems.

Each section of the paper presents simply stated lessons that were learned, often the hard way. If your own product situation is at all close to this, i.e., multiple versions on multiple platforms requiring support, then this paper is well worth the cost of the proceedings.

Software Design for Installability. Steve Simmons, Inland Sea.

Software installation is often added on at the end of the product development life-cycle as an afterthought. Steve's talk takes a system administrators point-of-view of what makes a package installable or not. (Requiring root privileges for installation is not a friendly requirement at most real sites with real sysadmins.)

Aspects of highly installable software packages are presented, with examples, and benefits to the software supplier of these approaches explained.

- Easier installation improves the likeliness of correct installation, reducing support costs.
- Respect for a customer site (with regard to policies, security, etc.) increases the value of the software to the customer, improving the likelihood of further sales.
- Ease of installation makes "test" runs and product samples more acceptable.

The examples are all based around the fictitious Whizzy-Wimp system, (What- you-see-is-what-I-mostly-programmed), but drawn from real war stories. Vendors names were removed to protect the guilty. An excellent presentation with an excellent set of guidelines.

Church's Lemma and Funkenhauser's Rebuttal; Designing a Model of the Software Process

David Tilbrook (MKS, Inc.) presented the Toronto development audience with a discussion of the inherent problems in the software management process. For those of you who know David, it was indeed another tirade of Tilbrookian proportions. (Yes, *that* talk.) The thing that still fascinates me, having seen David speak on several occasions, is that so many people are still amazed when these sorts of fundamental issues and concerns are laid out.

The kick-off point for the talk was based on a recent discussion David came across between two developers releasing product, that lead him to:

Church's Lemma: It's easy if you know how.
Funkenhauser's Rebuttal: No it's not.

The talk focused on the problems with software management, presenting the solution as a set of practices, such as:

- The complete separation of product source from the objects;
- A simple concise expression of the construction for a target is essential; this construction expression must be platform independent.
- Platform dependent configuration information is collected into a single place, such that it is easy to maintain and advance.

The audience was kept rolling along through David's presentation. There were lots of questions and discussion afterwards, spilling into the reception.

Summer '94 Conference Reports

The "Best" Awards

The Best Paper was awarded to Jeffrey C. Mogul, Digital Equipment Corporation, Western Research Laboratory for "A Better Update Policy."

The Best Student Paper award went to the co-authors: Trevor Blackwell, Kee Chan, Koling Chang, Thomas Charuhas, James Gwertzman, Brad Karp, H. T. Kung, David Li, Dong Lin, Robert Morris, Rob Polansky, Diane Tang, Cliff Young, John Zao, Harvard University for "Secure Short-Cut Routine for Mobile IP."

The Best Presentation was awarded to Matt Blaze, AT&T Bell Laboratories for "Key Management in an Encrypting File System."

Summaries of Selected Papers

by Pace Willisson
<pace@bsdi.com>

Protocol Failure in the Escrowed Encryption Standard

This talk was a surprise treat. The story about Matt Blaze's discovery of ways to "cheat" the Escrowed Encryption Standard (popularly known as the Clipper Chip) broke in major newspapers only a few weeks before the conference, and this talk replaced one of the originally scheduled invited talks. There were two parts: first a short description on the standard by Steve Bellovin, then a description by Blaze of the cheating techniques.

Bellovin said that the Data Encryption Standard (DES) has reached the end of its useful life, and that the U. S. government, via the National Institute of Standards and Technology (NIST), wants to establish a new, stronger standard for use with applications such as voice telephones, fax, and email. The algorithm selected was developed by the NSA, and is called Skipjack.

But, the government also believes that it should be able to listen in on any transmission. So, the NIST is also specifying a set of cryptographic protocols that must be used with Skipjack. The whole package is called the Escrowed Encryption Standard (EES). Under the EES, a Law Enforcement Access Field (LEAF) must be transmitted at the beginning of each message. This field, when combined with a record from a master data base, yields enough information to decode the whole message. In an effort to prevent LEAF-free use, the Skipjack algorithm will be kept secret, and will be deployed only in tamper-resistant chips. Furthermore, these chips will be sold only to approved vendors for use in approved products. The current chips are named Clipper and Capstone. (The name Clipper was an unfortunate choice – it is also a trademark of a company that has nothing to do with the EES.)

Another reason for keeping Skipjack secret is to avoid teaching new cryptographic techniques to people outside the NSA. One of the things that spurred the substantial civilian cryptographic research in the last 15 years was the publication of the DES in the late 70s. It's been speculated that the NSA doesn't want to see the same thing to happen again.

Coupled with the EES is a set of policies designed to protect the master data base. The information in the data base will be split up, so that a piece of information must be obtained from each of two government agencies to reconstruct a key.

Any federal, state, or local agency that has the legal authority to conduct a wiretap may request key components. The request must certify that the necessary warrants have been obtained, but the warrant itself need not be presented. There are provisions for keeping audit trails, but the policies explicitly state that they do not create any "actionable legal rights."

After Bellovin's introduction, Matt Blaze talked about his work. He said that representatives from the NSA and NIST have been traveling around the country trying to drum up support for the EES. One stop was a meeting at Bell Laboratories, which Blaze attended. Afterward, Blaze posted a summary of the meeting to *sci.crypt*. A few days later, he saw that he had email from someone in the government. His first inclination was to run, but then he went ahead and read the message. The writer thanked him for posting a good summary, and offered to give him a prototype encryption device. The intention is that you'll be able to buy this type of card at any computer store for about $100. The API for the card is called Mosaic, not to be confused with the popular WWW program, and the PC implementation of the API is called Catapult, not to be confused with the Catapult companies.

Blaze wondered if the elaborate schemes to force the use of a LEAF really worked. First, he observed that it is simple for the sender to encrypt the LEAF with another encryption algorithm, and then for the receiver to decrypt it before feeding it to the receiving chip. But, this isn't much stronger than using the other algorithm for the whole message. So, he concentrated was on ways to use pure Skipjack, but without the LEAF.

If both sides are willing to cheat, he found that the transmitter need not send the LEAF at all. The details are a little tangled, but the idea is that the receiver puts its chip in transmit mode and tells it to generate a LEAF for the current session key. It won't be exactly the same as the one the transmitter was supposed to have sent, but the only effect of the difference is to garble the first eight data bytes of the data transfer. So, the transmitter starts with eight null bytes, and the receiver throws away eight bytes of garbage.

Next, Blaze attacked the problem of building a "rogue" transmitter whose messages could be decoded properly by a standard receiver, but whose LEAF transmissions would be useless to a wiretapper. The LEAF packet contains a copy of the session key, encrypted by the unit key of the transmitter, plus a 16-bit checksum covering (at least) the session key and the initialization vector. The receiver doesn't know the transmitter's unit key, so it must depend on the checksum to validate the LEAF. But a 16 bit checksum implies that one in 65,536 random bit strings will appear to have a valid checksum, even though the rest of the packet is garbage.

So, the transmitter chooses a session key, then sets its chip to receive mode and starts feeding in random 128 bit strings. In about 65,536 tries, one of these bit strings will appear to be a valid LEAF for the session key, and the chip will unlock. The transmitter then sends this LEAF in place of the true one, puts it chip back into transmit mode, and starts sending data.

After the original paper was released, the NIST responded and said, "the attacks described by Dr. Blaze were fully understood at the time of initial chip design," but also said "[The] NSA already has incorporated additional security safeguards into the production TESSERA cards to protect against the kinds of attacks described by Dr. Blaze."

Blaze's complete paper is available at *ftp://research.att.com /dist/mab/eesproto.ps*. Other EES-related material is available via ftp and WWW at *cpsr.org*.

A Toolkit and Methods for Internet Firewalls

Marcus Ranum and Fred Avolio of Trusted Information Systems talked about their application-level firewall toolkit. Instead of implementing the firewall by carefully filtering packets in a router, their system uses a workstation that has separate interfaces on the internal and external networks. The workstation does not forward packets at the network level – instead, a proxie process runs for each instance of an application. This allows considerable flexibility in selecting access policies (for example, the ftp proxie can allow import of files, but prevent export), while still using unmodified clients and servers. The only change from the user perspective is that hostnames may have to be specified in unusual places.

For example, to telnet from outside to a protected host, you would first telnet to the firewall. This starts the telnet proxie, which prompts for the name of an internal host. Then, it makes appropriate policy checks (e.g., allowed source host), and perhaps engages in a challenge/response user authentication. If everything is okay, it starts a new telnet connection on the internal network, then sticks around to forward data in both directions.

The proxies have to know a fair amount about their protocols (spoofing telnet option negotiations, for example), and sometimes have to go through extra contortions to avoid confusing existing clients. But, this work has been done for the most popular services, and other proxies are being added. The extra overhead apparently is not too high: Ranum claims that in a typical installation using a modern workstation as the firewall, the bandwidth of a 1.5 megabit/second link is more constraining than the speed of the proxies.

The toolkit has been designed explicitly with the thought of verifying its safety. The proxie processes are fairly small, limit file system accesses and run as unprivileged, chroot'ed programs. Also, all unnecessary subsystems are removed from the firewall machine. While this is certainly not airtight, it does dramatically reduce the amount of code that has to be trusted.

The components of the firewall toolkit can be used standalone and in various subsets. Ranum said that, thanks to the public release of source code, users have contributed substantial enhancements to the system.

SNP: An Interface for Secure Network Programming

In the next talk, Thomas Woo, et al., of the University of Texas at Austin considered the problem of adding security services to general applications. (It will be interesting to see how the tension between security aware applications and application-level firewalls plays out . . . neither of these talks addressed this issue.)

Until now, fancy authentication mechanisms, such as Kerberos, have only been widely available in a few programs, such as `rlogin` and `rsh`. The recently published GSS-API provides the primitives needed for secure authentication, integrity and confidentiality, but it is pretty complicated. The SNP library is an attempt to put an easy-to-use face on the GSS-API.

Woo claims that the net result is that the typical sockets-based program can be converted to use GSS-API features by simply adding the prefix "`snp_`" to all of the socket-related calls, and adding one new call just after creating the socket.

Behind the scenes, each data connection is augmented by a separate TCP control connection for communication between the SNP libraries at each end. The performance hit is dominated by the software cryptographic functions, and is substantial in the current implementation. It takes slightly more than two seconds to establish a connection with a SPARC 10/30, but 90% of the time is consumed by the relatively non-optimized RSA and DES cryptographic functions. The data transfer is not substantially affected (unless you turn on encryption for that too).

An Efficient Kernel-Based Implementation of Posix Threads

Robert Alfieri of Data General questioned the conventional wisdom that a kernel implementation of threads would be too slow due to system call overhead. He said that the attempts to implement mostly user level threads have lead to complex designs requiring elaborate synchronization between the kernel and user-level states. He addressed the heart of the problem: why are system calls slow? He found that if you abandon many of the traditional semantics of the system call, you can implement very fast "kernel function calls" for doing simple things. Among the simplifications, a kernel function call is only allowed when the process is not being debugged, and it may not get any page faults. Furthermore, the kernel code runs with interrupts off. If a kernel function call cannot complete within these rules, it is "promoted" to a full fledged system call by appropriate stack shuffling.

With this in place, he implemented a thread system where the kernel knows about every thread and every transfer of control. This allows widespread simplifications, even in places such as the debugger and `ps(1)`.

Although the goal was to be "not much worse" than user-level threads, actual performance measurements showed the new implementation to be better in almost all cases.

Using OS Locking Services to Implement a DBMS

Andrea Skarra of AT&T Bell Laboratories wondered if the user-level lock manager used by most serious data base systems is really necessary. Instead, perhaps file locking as provided fcntl could be used to implement a two-phase locking transaction system. She did a series of experiments on several kernels, but found many problems. None of the kernels made any attempt to prevent new read locks from being set while a write lock request is pending. This leads to write starvation. Similarly, a large read lock request may be delayed indefinitely by a series of small read locks on overlapping data. One implementation couldn't upgrade locks properly, and no implementation prevented accidental downgrades. A number of implementations had small limits on the number of locks that can be set at one time; one implementation stopped setting locks after a while, but didn't bother returning errors. Another implementation crashed the kernel after making a few thousand locks.

She believes that if `fcntl` could be extended in a few backward compatible ways, and if the present bugs are fixed, then it would be possible and desirable to use it instead of a user-level lock manager.

The Slab Allocator: An Object Caching Kernel Memory Allocator

Jeff Bonwick of Sun Microsystems described his new kernel memory allocator for SunOS 5.4. It sounds like the project started with some preconceived notions that certain objects had so many constant fields that allocation took too long. So, the first change was to keep such objects on a freelist, and only initialize those fields the very first time. Once he got into doing serious measurements of the memory allocation system, he wound up rewriting it.

I don't think that the results support the idea that avoiding object reinitialization helps much. However, his new allocator does substantially reduce kernel memory fragmentation, leaving more memory available for user processes. In fact, two memory-bound benchmarks showed 10 to 20 percent shorter runtimes with the new allocator. His other benchmarks showed a few percent improvement, probably due to a factor of two improvement in the basic malloc/free time.

The key idea is to have separate freelists for different size objects, and to bunch objects of the same size together in memory. His system has about 30 freelists for generic objects ranging from 8 bytes to 9 kilobytes in 10 to 20 percent increments, plus a few object-specific freelists. The time to allocate an object is reduced, since it isn't necessary to search though a lot of objects of the wrong size; and fragmentation is reduced, since allocating 40 bytes, freeing them, then allocating 32 doesn't leave a useless 8 byte hole.

The most surprising discovery is that an allocator that works in powers of two winds up wasting a couple of percent of the overall system performance due to cache line contention. Most objects have a few heavily used fields, and if all such objects start at similar offsets, these fields will map to a small subset of the available cache lines. His allocator not only avoids shifting objects to power of two boundaries, but also explicitly distributes wasted space between the front and back of the pages so that objects start at every possible offset.

Sawmill: A High Bandwidth Logging File Systems

The Berkeley RAID group continues to try to eke out the maximum performance from cheap disk drives. This latest work, by Ken Shirriff, explores using a log file system along with a custom disk controller that can communicate directly with the network interface.

The result is an order of magnitude increase in performance over a conventional RAID. The read bandwidth tops out at 21 megabytes per second, and write at 15 megabytes per second in their present implementation, about 70% of maximum possible.

The key to achieving high bandwidths is to avoid cases where the file server has to handle a piece of data more than once. For example, in a traditional log file system, a write request first gets entered into a block cache. Later, when it is time to write a segment, the dirty blocks are scanned, sorted, and arranged for writing to the disk. In Sawmill, there is no server cache, and the position for write data is assigned at the time the write request arrives. When the write buffer becomes full, it can be written directly to disk.

There is still plenty more work to do – the cleaner has yet to be implemented, so its performance impact is unknown. Also, they did not have a client that could handle such high bandwidth transfers, so the performance figures are estimates based on measuring internal file server operations.

Application-Controlled File Caching Policies

Pei Cao, Edward W. Felten, and Kai Li of Princeton University studied ways for an application to help the kernel choose what blocks in a file cache should be replaced. The main problem is to prevent a process from inflating its working set size by making poor choices. They formulated a precise statement of this goal, and then found a simple, though subtle strategy to achieve it. You'll have to look at the paper for details.

It seems like the easiest way to pass these hints to the kernel would be to offer a few replacement algorithms (MRU, lowest block number, etc.) and the user process would select one with an fcntl. There are other possibilities, ranging all the way to making an upcall during each cache miss.

They simulated their design using file system traces and three simple replacement algorithms. For linking a kernel, the cache miss ratio dropped from about 50% to 25%. Several other programs showed significant, though less dramatic improvements.

Reducing File System Latency Using a Predictive Approach

James Griffioen and Randy Appleton of the University of Kentucky explored what would happen if you get aggressive about doing file read ahead. Traditional read ahead notices when you've read the first two blocks of a file, then reads the third. But this means you are sure to have to wait for a couple of blocks the first time you read a file. Attempting to do the read ahead at open time doesn't help much. The average time from open to first read is about 3 milliseconds – much less than the latency of a disk drive.

So, they wondered how to do read ahead before open, but without having to change any application programs. They observed that there are many cases where opening one file implies that another particular file is likely to be opened next. While compiling gcc, for example, an open of "`config`" is always followed by an open of "`tm.h`", and that is usually followed by an open of "`alloca.h`". So, they imagine extending the inode to have a list of "probable next files." When a file is opened, then the kernel would update the probabilities in the inode of the previously opened file. Whenever the system isn't otherwise busy, it would prefetch the next most likely file, out to some lookahead limit.

They simulated the performance of this system using file system traces. They found the biggest benefit to be for small size caches, with up to a 280% reduction in the miss rate. For all cache sizes, prefetching beat LRU by at least 10%.

A Text Retrieval Package for the UNIX Operating System

Liam Quin of SoftQuad Inc. needed a full text retrieval system, but couldn't afford the commercial packages that cost many tens of thousands of dollars. So, he wrote his own, based on ndbm. It features compact index files (only 1/2 to 1/4 the size of the input text) and can do accurate phrase matching. The text need not be kept online after the index is built. With it, you can find a phrase in the 30 megabytes of new.answers in less than one second, compared to three minutes using `grep`.

He talked about the many strategies he used to keep the size of the index down. He asserted that in just about any body of text, be it netnews, the Bible, or whatever, that most words occur one time. A handful may appear thousands of times, but practically all occur fewer than 10 times. Thus, his main data structure is optimized for storing words that occur a few times, and has provisions to overflow when necessary. Another optimization is to store numbers as variable length byte strings, and to store a series of numbers as a starting value, then a list of deltas.

Quin spent some time talking about his philosophy of text retrieval. He thinks it is very important for the index to have complete word order information to reduce the number of "false drops" that can't be discovered until the original documents are consulted. He also says that he hates "stop lists," which omit common words from the index. His packages supports them, but since the delta coding of file numbers works so well, you don't really need them. He said that the acid test for a retrieval system is to look up "To be, or not to be" in Shakespeare.

Summary of "Technical Executive Summaries"

by Jerry Peek, O'Reilly & Associates
<jerry@ora.com>

This session was a series of eight short talks by seven speakers. Each talk covered the latest developments in a particular area.

Cryptography Issues

Phil Zimmerman started his talk with a news flash. MIT Press will publish the full source code for PGP (his Pretty Good Privacy encryption software) as a book. Phil has been under investigation (but not indicted) for possibly violating U.S. laws against exporting encryption software in electronic form. But there are no export controls on source code printed in books! [*Editor's Note: see related article on page 8*]

RISC vs. CISC

Rik Farrow compared the Intel Pentium CISC chip to a whole series of RISC chips. Although the Pentium is still compatible with the ancient 4004 terminal controller chip (!), his SPECint92 benchmarks showed the Pentium beating *all* the RISC chips. The floating point performance, which is increasingly important, varied quite a bit chip-to-chip; but the SPECfp92 number for Pentium was still respectable. Rik summarized the Pentium architecture and predicted that Intel's job will get tougher as other manufacturers' RISC chips speed up; the Pentium chip already has about ten times as many transistors as its competitors.

Hot Chips, Cool Code!

Mike Smith listed a series of hot topics in microprocessor architecture: low-power design; multimedia extensions (for MPEG, etc.); embedded designs; PDAs (personal digital assistants); DSPs (digital signal processors); and MP (multiprocessing) chip sets.

Most of the talk focused on the instructions-per-cycle issues in Superscalar Processors: a detailed set of vue-graphs that covered the overall topic and some implementation information. Multiple-issue processors execute instructions in parallel; they can have improved performance over traditional pipelining, but there are (obviously) some tough problems. Current out-of-order issue processors rely on hardware techniques for determining when instructions can be issued out-of-order. Multiple-issue processors are starting to use VLIW (Very Long Instruction Word) techniques and compiler control of out-of-order issue. (And no matter what technique you use, the actual instruction stream becomes very messy.) Only two processors today use out-of-order issue, though more are coming. Mike believes that all these processors will exceed 200 SPECint94 with an out-of-order issue of 4-6 instructions. [Thanks to Mike Loukides for help with this summary. – JP]

Recent Trends in Intellectual Property

Most of Dan Appelman's talk focused on U.S. patent law.

Last year, there were 8400 software patent applications filed; 3600 patents (most from previous years) were issued. A software patent may take from 2 to 10 years to approve

The Patent and Trademark Office (PTO) now routinely grants software patents; they didn't always.

PTO has acknowledged that many of their examiners are inexperienced in software patents – and that examiners don't have enough examples of "prior art" to guide them. They're evaluating the patent process to see if it can be made more like other countries'. For example, the U.S. PTO might start to disclose claims before a patent is issued – instead of keeping the application secret until it's approved.

Three particularly interesting patent cases:
- Compton's very broad patent on text retrieval has been overturned.
- The Software Advertising Corporation patented the incorporation of advertising into software (!). PC Dynamics Corporation got the patent re-examined with an innovative technique: Instead of using lawyers, they hired a public relations firm and lobbied Congress.
- Stak Electronics won a judgement against Microsoft for infringing Stak's patent on file compression.

Dan's firm represented BSDI in UNIX System Laboratories' suit against BSDI. Although the settlement terms can't be disclosed, Dan mentioned an interesting point. During the suit, USL tried to get an injunction to stop BSDI's software distribution; USL alleged that the distribution violated copyrights and trade secrets. The judge denied the injunction, ruling that there were no copyrights or trade secrets left in UNIX 32V! That judgement had to be vacated, though, when the suit was settled out of court.

Research in Networking

David Clark's Megabit-speed :-) presentation covered a lot of developments in ten minutes.

Real-time networks don't require ATM. A setup protocol called RSVP lets the receiver say that they especially want to receive certain packets (and will pay more money to receive them). The changes needed for RSVP have been made and tested on routers.

ATM isn't the total answer; David feels that it won't displace IP. ATM will be a "fancy subnet technology used in longlines and LANs." In longlines, there'll be a fight with the established Frame Relay; it isn't clear that ATM will be used for voice (part of its design spec) at all! There's lots of action in ATM LANs; 155 mb/s switches cost less than $1000/port now.

Five years ago, Internet security was a topic that bored and angered a lot of people. Now it's being taken seriously. Attacks are more sophisticated. Firewalls are more common. Denial-of-service attacks are being analyzed and dealt with.

Work on IPng, the next-generation IP that will expand the current 32-bit address space, is coming along.

The "killer Internet apps" are WWW and Mosaic, not email. For example, the MIT Library is scanning 1 Gigabit of new material online every day. Also, systems for online commerce and electronic money are moving onto the Internet.

The Future of the Internet

David Clark's crystal ball showed:

The Internet is going commercial. The NSFNET backbone is being replaced by commercial services. The good news: you won't pay much more because the longlines service is a small percentage of your total usage cost. And the AUP (Acceptable Use Policy) will go away! The bad news: There's fear and confusion. Lots of competing interests want to make money.

His best guess for 155 Mb/s ATM is: Real Soon Now, at US $3000 per month. This will be *cheap* for big institutions; bulk transmission is not expensive. What's expensive is splitting this bulk bandwidth into small chunks.

NII, the US National Information Infrastructure, is coming. Two of the main competing visions are:
- 500 channels of TV
- An open market of services and ideas

But David pointed out that networks aren't built out of visions; they're built with hardware and money. If the USA will have subscriber loops wired into everyones' homes, who will pay the estimated $50 billion cost – and under what circumstances?

His report, *Realizing the Information Future*, discusses the commercialization of the Internet and the architecture of the NII. You can get it from National Academy Press, *{www,gopher,ftp}.nas.edu* or by calling (in the US and Canada) 1-800-624-6242 and paying money.

Financials of High-Tech Companies

Michael Carmen of State Street Finanacial Services started by defining PCMCIA as "People who Can't Memorize Computer Industry Acronyms." :-) He says the biggest problem faced by financial portfolio managers is understanding companies' products.

There are three things he looks for in companies: very strong management teams that can build an organization and execute great ideas; good technology; and earning money.

Product areas he likes include: semiconductors (there's an increasing demand but a limited supply); client-server, and

Thanks for the Prizes!

USENIX would like to thank SunSoft for its generous $5000 donation supporting the Grand Prize and Finalist Prizes for U-Word Jeopardy! and the prizes for all other contest as well.

USENIX also thanks BSDI for donating a complete BSD/386 system for the Grand Prize winner, and O'Reilly & Associates for its donation of 18 books for contest prizes. All contestants received two O'Reilly books of their choice.

U-Word Jeopardy:

Jeopardy winner: Guy Harris, Network Appliance Corp.

Jeopardy finalists: Ron Gomes, Morgan Stanley & Co., David Korn,AT&T Bell Laboratories

Jeopardy semi-finalists: Clement T. Cole, Locus Computing Corp.; Peter Honeyman, CITI, University of Michigan; Jim Reid, UKUUG; Andrew Tannenbaum, Clam Associates; Christopher Torek, Berkeley Software Design, Inc.; James W. Williams, Hughes STX

Winner of Card Deck contest: Brad Templeton, Clarinet; runner-up Jeff Mogul, Digital Equipment Corporation, Western Resarch Laboratory

Board Meeting Summary

by Ellie Young
<ellie@usenix.org>

Below is a summary of the actions taken at the regular quarterly meeting of the USENIX Board of Directors which convened in Boston, Massachusetts on June 5, 1994.

Attendance: Adams, Allman, Christiansen, Geer, Grob, Hume, Johnson, Nemeth, O'Dell, Rose, Shein. Appelman, DeMartini, DesHarnais, Klein, Knight, Young. Haemer, Kenyon, Biel-Nielsen, Seltzer, Walli, Zwicky.

Boston Conference

Margo Seltzer reported that the program committee had received 107 submissions of which 27 were accepted. She thanked SunSoft, BSDI, and O'Reilly & Associates for their donations of prizes for the U-Word Jeopardy and Bingo contests. Nemeth thanked DEC for loaning equipment to do the various broadcasts over the Mbone. It was estimated that 1100 would finally attend.

UNIX Applications Development Symposium

It was agreed that the board would consider a proposal from Stephe Walli to sponsor another event on this topic.

C++ Conference/COOTs

The program committee of the recent C++ conference had indicated its interest in having this conference replaced with one that would focus on practical object-oriented technologies (COOTS). Young reported that Vince Russo would serve as program chair, and it would be held in June, 1995 (see p. 49 for the Call for Papers).

Mobile & Location-Independent Computing Symposium

The proposal from Jim Rees to sponsor a second symposium on this topic next Spring was accepted.

UNIX Security Symposium

Fred Avolio had agreed to serve as program chair. A committee of past organizers and board members had also met to discuss the timing and program for this event. It was unlikely that it could be held alongside the LISA conference in the fall because of other security- related events also being held in that timeframe. Young would work with Avolio on plans to hold it in June, 1995.

LISA '95 Program Chair

The committee's recommendation to accept the proposal to chair from Tina Darmohray and Paul Evans was accepted.

Executive Director's Report

Young asked for input on the USENIX WWW page. She would submit a proposal/plan for putting up papers from our proceedings, which would be accessible by our members. She would also look into online ordering. It was suggested that we include past programs and abstracts, expand the calendar to include links to the EurOpen national groups, JUS, and AUUG. Young reported that the 4.4BSD Manuals and CD-ROM which we were co-publishing with O'Reilly would be available in July (see page 4).

SAGE

Zwicky reported that they were working to get more documents out in view of the success of the jobs description booklet. She also felt that the SAGE co-sponsorship of SANS and UniForum events went very well, and plans were being made to do so again in 1995.

EurOpen

Kim Biel-Nielsen, the chairman of EurOpen, reported that they would be publishing a quarterly publication which would include the best articles from other national group's newsletters. Simon Kenyon would be working on network

services ((setting up WWW) and conferences (they hoped to launch another one with a partner in 18 months). They would like to continue cooperation with UniForum and USENIX (perhaps doing a joint conference) and various other avenues would be explored in the near future.

UniForum

Johnson reported that UniForum wanted to do more cooperative ventures and had invited him to serve on their 1995 conference planning committee. It was agreed that we should proceed with organizing a track and/or tutorials.

20th Anniversary Celebration

The Board asked Young to come up with a budget for consideration at the next meeting to celebrate USENIX's 20th anniversary at the Winter conference in New Orleans.

Journal Report

The proposal from MIT Press to raise non-member subscription rates for 1995 from $65 to $75 was accepted. O'Dell reported that the first two issues for 1994 were out and the balance of this year's volume was on track. He announced that Dave Presotto would officially be taking over as the journal editor effective with Volume 8 for 1995. O'Dell was formally thanked for his contribution was starting and running the journal.

USENIX Board Committees

The following board committees were re-structured: Executive: Johnson, Adams, Allman, Grob. Awards/Scholarships: Adams, O'Dell, Scherrer, Rose. Promotion: Johnson, Grob, Knight, DesHarnais, Young. Publications: Presotto, Young, Christiansen. Tutorial Review Committee: Klein, Young, Geer, Honeyman, Farrow, Hein, Grob.

Next Meeting

It will be held in Chicago on October 11 and 12, 1994.

Letter to the Editor

About the Jan/Feb 94 "Review of BSD/386"

by John Lockwood
<lockwood@ipoint.vlsi.uiuc.edu>

Don't get me wrong. I don't want to start a flame war – this already has raged through the net.

I couldn't help but read the "Review of BSD/386" article and not think of how my experience with installing Linux was almost exactly the same as Lou Katz's experience with installing BSD/386. Taking a Slackware-or-SLS distribution (even more easily done from net, but can be done from CD), I was also able to enjoy X11, gcc/g++, NFS, telnet, LaTeX, xdvi, ghostview, SLIP, etc., all "out of the box." But the cost: $0.00.

It's important that installing UNIX is so easy – that makes it attractive for a large number of people. But when it's also available for free, all the better.

(Editor's Note: We are always soliciting full reviews of software. . . RK)

Executive Office Staff Addition

Zanna Knight, the USENIX Marketing Director and Vendor Display Coordinator, comes to us from Vanstar Corporation, formerly ComputerLand. Her responsibilities at Vanstar included the marketing, program development and execution of their conferences, trade shows, seminars and training to the Fortune 2000 companies, end-users, vendors and staff. She has a B.A. from Queens College, City University of New York and an M.A. in French from Middlebury College which afforded her the pleasure of living in France for two years. Her special passion is travel and she once went by Landrover from Paris to New Delhi. Welcome aboard!

Community News

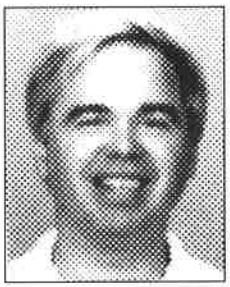
Congratulations to:

David Fiedler who proudly announces the birth of his son, Steven Jacob Fiedler on March 14, 1994 (5 pounds 12 ounces, 19 inches). Steven follows five years after David and Susan's first (Scott); and

Mike O'Dell and Judy Grass who tied the knot in February.

Jaap Akkerhuis who was married to Nadja Saitseva in Sergiev Posad (formerly Zagorsk), Russia in July.

Board of Directors



Stephen Johnson
President

Eric Allman
Vice President

Lori Grob
Secretary

Rick Adams
Treasurer

Tom Christiansen
Director

Dan Geer
Director

Andrew Hume
Director

Greg Rose
Director

Staff

Ellie Young
Executive Director

Zanna Knight
Marketing Director Exhibits

Carolyn S. Carr
Publications Manager

Diane DeMartini
Administrative Assistant

Toni Veglia
Administrative Assistant

Lilia Scott
Administrative Assistant

Judith DesHarnais
Conference Planner

Karen Krueger
Administrative Assistant

Daniel Klein
Tutorial Coordinator

SAGE
THE SYSTEM ADMINISTRATORS GUILD
NEWS

Letter from the SAGE President

by Elizabeth Zwicky
<zwicky@corp.sgi.com>

Since this seems to be the time to celebrate anniversaries, let me point out that while UNIX and the Arpanet celebrate 25th anniversaries, and USENIX celebrates a 20th, SAGE is reaching the ripe age of two (old enough to race, for a horse), SAGE-AU is turning one, and BAYLISA became a venerable four in January. A bit early for commemoration, but a source of pride to us all nonetheless.

Several of the working groups have projects in progress. The working group on local groups is working on a pamphlet about starting up local groups; the policies working group is working on a document which will help people develop their own policies; and the security and online information distribution groups are adding security information to SAGE's WWW server.

Local groups continue to spring up, including $GROUPNAME in New Jersey, New York Systems Administrators (NYSA) in New York City, North Carolina System Administration Interest Group (NCSA) in North Carolina, and English Bay LISA in Toronto (for more information, contact Peter VanEpp, *<vanepp@sfu.ca>*). A group is also starting up in Texas; contact A. Bryan Curnutt, *<bryan@stoner.com>*. For other contact information please see the Local Groups section in the back of this newsletter.

Everything SAGE does is a product of volunteers (with finishing touches provided by a small but dedicated crew of professionals in the USENIX office). What we need most is people who're willing to help. If you're willing, but you don't know where to get started, let me know; I can usually find something useful for you to do, which may bring you small but pleasant amounts of fame. (I'm afraid fortune is unlikely.)

From the Editor

by Bryan McDonald
<bigmac@usenix.org>

As Elizabeth mentions above, volunteer effort drives most of the developments in SAGE. Since I have the space and the lead in, I would like to add my plug as well. I am always looking for people interested in writing for the SAGE newsletter, either to review books, software or other interesting items, or simply people interested in sharing their experiences earned as they moved through this field. You would be suprised how many of you have very relevant stories to pass on that would help both the old hands and those joining our ranks. If you are not sure you have anything to say, but are interested in writing, drop me a line and we will find out just how much you do have to say to the community.

SAGE, the System Administrators Guild, is dedicated to the advancement of system administration as a profession. Now just two years old, SAGE's membership continues to increase steadily, and there is growing recognition of SAGE as a representative in system administration issues. SAGE brings together system and network administrators for:

- professional and technical development,
- sharing of problems and solutions,
- communicating with users, management, and vendors on system administration topics.

SAGE NEWS EDITOR
- Bryan McDonald
  <bigmac@usenix.org>

SAGE BOARD OF DIRECTORS
- Elizabeth Zwicky, President
  <zwicky@usenix.org>
- Paul Evans, Secretary
  <ple@usenix.org>
- Peg Schafer, Treasurer
  <peg@usenix.org>
- Paul Moriarty
  <pmm@usenix.org>
- Pat Parseghian
  <pep@usenix.org>
- Steve Simmons
  <scs@usenix.org>
- Pat Wilson
  <paw@usenix.org>

SAGE WORKING GROUPS

GROUP	CHAIR
sage-certify	Arch Mott
sage-edu	Paul Evans
sage-ethics	Ed Gould
sage-jobs	Tina Darmohray
sage-locals	Rene Gobeyn
sage-online	Mark Verber
sage-policies	Lee Damon
sage-pubs	Bryan McDonald
sage-security	Arnold & Laura de Leon
sage-stds	Janet Frazier
sage-vendors	Dave England

You can contact these groups via email: *<their name@usenix.org>* for example, *<sage-certify@usenix.org>*.

SAGE DISCUSSION GROUPS
sage-managers
sage-outreach
sage-pt

SAGE ONLINE SERVICES
Email server:
majordomo@usenix.org

FTP server:
ftp.sage.usenix.org

WWW URL:
http://www.sage.usenix.org

SAGE SUPPORTING MEMBER
Enterprise Systems Management Corp.

Highlights from the June SAGE Board Meeting

by Paul Evans
<ple@usenix.org>

The SAGE board met at the USENIX Conference in Boston on Tuesday June 7, 1994. Present were Elizabeth Zwicky, president, Peg Schafer, treasurer, Paul Evans, secretary, Steve Simmons and Pat Wilson, board members, Tom Christiansen and Greg Rose from the USENIX board, and Ellie Young, Judy DesHarnais, Zanna Knight and Diane DeMartini from the USENIX Association offices.

Location of Future LISA Conferences.

The board reviewed possible future locations for the LISA Conference in the 1996 and 1997 time slots. LISA 8 (this year) will be at the Town and Country in San Diego; LISA 9 (1995) will be in Monterey again. Some board members have expressed concern about LISA having been on the West Coast, and particularly in the state of California, for five years in a row. With the decision by the USENIX board to discontinue the summer technical conferences, the SAGE board had the option of picking up the slot held for the June 1997 summer conference in Chicago. This left the location for the 1996 LISA conference undecided. The board has asked Judy DesHarnais to look into the possibility of locating the conference for that year in Portland or Seattle.

Nominating Committee

The SAGE board elections will be coming up again in the fall, with three seats on the board of directors open. The nominating committee was formed, and includes Greg Rose, Pat Wilson, and Elizabeth Zwicky. More members of this committee may be recruited from the SAGE membership. Members of the nominating committee are responsible for ensuring that an adequate number of candidates run in the election, and that the candidates, their family members, employers, and other interested parties understand the demands that will be made on board members. The members of the nominating committee are not eligible to run for the board themselves.

Conferences

SAGE has had an ongoing relationship with the SANS (Systems Administration and Network Security) Conference which is held every April in Washington, D.C. SAGE has in the past provided several member of the SANS program committee; Peg Schafer and Pat Wilson volunteered to serve on the SANS program committee for next year. Planning is already underway for the LISA 9 conference, scheduled to be held in September 1995 in Monterey. Tina Darmohray and Paul Evans were selected as program co-chairs for that event.

This year for the first time, SAGE and USENIX provided a tutorial track on security and systems administration management issues at the UniForum conference. SAGE hopes that this effort will make a new audience aware of SAGE as an organization and other conferences that SAGE is innvolved, with like LISA and SANS. Elizabeth Zwicky has volunteered to coordinate with the staff the USENIX/SAGE track at the UniForum 1995 conference.

Publications

In the past, SAGE has provided a free copy of any new publications it released to its members. The board has decided that, in the future, new members will also receive a copy of the most recent publication, which at this time is "Job Descriptions for System Administrators." In another publications-related action, the SAGE board re-appointed Bryan McDonald SAGE editor of this newsletter. Congratulations on your outstanding work, Bryan!

Awards

SAGE will once again present at the LISA 8 conference in San Diego its Outstanding Achievement award. The award will go to someone who's contributions, either technical or professional, to the system administration community over a number of years merit special recognition. The SAGE board set up awards committee consisting of Paul Evans, Steve Simmons and Pat Wilson to invite suggestions and make a recommendation to the board by early July.

Call for Nominees for Election to SAGE Board of Directors

by Pat Wilson
<paw@usenix.org>

SAGE is accepting nominations for 3 new members of its Board of Directors until October 14, at noon, PST. Anyone interested in running for the SAGE board should send his or her name and telephone number and a brief statement to the nominating committee via email at:

sage-nomcom@usenix.org

You can also send U.S. Mail to the SAGE Nominating Committee care of:

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710

The nominating committee will gather the candidates' names and contact each of them before the election takes place.

In this election, directors will be chosen for 2 year terms (beginning January 1, 1995) and will join returning Board members Paul Evans, Paul Moriarty, Pat Wilson, and Elizabeth Zwicky (elected last year to two year terms which began Jan 1, 1994). The SAGE Board chooses its own officers after each general election (every year). At the USENIX LISA Conference, to be held September 19-23, 1994 in San Diego, CA, there will be a candidates' forum to enable candidates to introduce themselves and talk about the issues. Prospective board members unable to attend the LISA conference will be able to submit a position paper to this forum. All candidates will be expected to respond for publication to a set of questions presented by the Nominating Committee. There will, in addition, be an on-line forum (most likely an archived mailing list) to enable SAGE members to pose questions to the nominees.

The new board will take office in January, 1995, with their first meeting being held at the Winter USENIX conference in New Orleans, Louisiana. Current estimates indicate that the new board will have at least 2 face-to-face meetings a year, one each at LISA and at the (winter) USENIX technical conferences, and other meetings via teleconference.

If you have questions about the nominating process, or what Board membership entails, please contact a member of the Nominating Committee:

Pat Wilson <*paw@northstar.dartmouth.edu*>
Greg Rose <*ggr@usenix.org*>
Barry Wick <*wick@ssdevo.enet.dec.com*>
Elizabeth Zwicky <*zwicky@corp.sgi.com*>

SAGE-AU Code of Ethical Conduct

by Kate Lance
<*clance@cs.newcastle.edu.au*>

[From the SAGE editor: SAGE-AU has been devoting lots of energy to the discussion of codes of conduct. By the time this article is printed, the SAGE-AU conference in July should be over, and this will have gone beyond draft stage, but I wanted members of SAGE here in the U.S. to be aware of the great effort and work that SAGE-AU is putting into our global community. – Bryan]

Members of SAGE-AU have been discussing ethics via a mailing list since January this year. The outcome is a Code of Ethical Conduct which will be presented at the annual meeting of SAGE-AU in Perth, Western Australia, in July. The code shown below is the current draft – we intend to have the final version ready before the end of June. If you would like to offer any comments, please send email to Kate Lance <*clance@cs.newcastle.edu.au*>.

In a very short period of time computers have become fundamental to the organization of societies world-wide; they are now entrenched at every level of human communication from government to the most personal. Computer systems today are not simply constructions of hardware – rather, they are generated out of an intricate interrelationship between administrators, users, employers, other network sites, and the providers of software, hardware, and national and international communication networks.

The demands upon the people who administer these complex systems are wide-ranging. As members of that community of computer managers, and of the System Administrators' Guild of Australia (SAGE-AU), we have compiled a set of principles to clarify some of the ethical obligations and responsibilities undertaken by practitioners of this new profession.

We intend that this code will emphasize, both to others and to ourselves, that we are professionals who are resolved to uphold our ethical ideals and obligations. We are committed to maintaining the confidentiality and integrity of the computer systems we manage, for the benefit of all of those involved with them.

No single set of rules could apply to the enormous variety of situations and responsibilities that exist: while system administrators must always be guided by their own professional judgment, we hope that consideration of this code will help when difficulties arise.

Note: In this document, the term "users" refers to all people with authorized access to a computer system, including those such as employers, clients, and system staff.

SAGE-AU Code of Ethics

As a member of SAGE-AU I will be guided by the following principles:

1. Fair Treatment

I will treat everyone fairly. I will not discriminate against anyone on grounds such as age, disability, gender, sexual preference, religion, race, or national origin.

2. Privacy

I will access private information on computer systems only when it is necessary in the course of my duties. I will maintain the confidentiality of any information to which I may have access. I acknowledge statutory laws governing data privacy such as the Commonwealth Information Privacy Principles.

3. Communication

I will keep users informed about computer matters that may affect them–such as conditions of acceptable use, sharing of common resources, maintenance of security, occurrence of system monitoring, and any relevant legal obligations.

4. System Integrity

I will strive to ensure the integrity of the systems for which I have responsibility, using all appropriate means – such as regularly maintaining software and hardware; analyzing levels of system performance and activity; and, as far as possible, preventing unauthorised use or access.

5. Cooperation

I will cooperate with and support my fellow computing professionals. I acknowledge the community responsibility that is fundamental to the integrity of local, national, and international network resources.

6. Honesty

I will be honest about my competence and will seek help when necessary. When my professional advice is sought, I will be impartial. I will avoid conflicts of interest; if they do arise I will declare them.

7. Education

I will continue to update and enhance my technical knowledge and management skills by training, study, and the sharing of information and experiences with my fellow professionals.

8. Social Responsibility

I will improve my understanding of the social and legal issues that arise in computing environments, and I will communicate that understanding to others when appropriate. I will strive to ensure that policies and laws about computer systems are consistent with my ethical principles.

9. Workplace Quality

I will strive to achieve and maintain a safe, healthy, productive workplace for all users.

Please Unsubscribe Me from this List . . .

by Bryan McDonald
<bigmac@usenix.org>

I have been seriously working on the Internet for 6 years now, and it amazes me sometimes. When I first got heavily involved, I was emerging from the dungeons of the UC Davis MIS shop where I worked as a glorified gopher. I had this vision of thousands or millions of intelligent, learned people all plying their trade on the Net (this was well before someone in Washington coined the term Information Superhighway). I had certain expectations about these people: after all, this network was populated by some of the best minds in the world, gathering to create a new frontier. I guess that image set up expectations too high to meet and too pervasive to die, because I am always astounded and disheartened when another round of idiocy makes its way into my inbox.

For example, someone at a large vendor site recently got hold of a large list of email addresses, created a mailing list named foo-invite, and then sent a message to it, advertising a free, online information service. So, mildly annoying, not only because I already subscribed to said service, but also because I usually prefer to get my junk info in news groups or in my snail-mail. However, it is not as if they were trying to sell me something, it was a reasonable service, and I just removed the message.

No, this is not what prompted me to write this article. What followed was the culprit. Following this invitation, I began wading through a stream of abusive email all aimed at flaming the sender for throwing junk email at them. After all, they already got too much email and their time was too valuable to waste removing it. Of course, the flamers did not have the common courtesy to send it to the author of the

message, instead sending it back to the entire mailing list, which I am sure spanned hundreds, if not thousands, of people across the Internet.

Other examples abound. Shortly before the above incident I got a message on a mailing list covering a certain network device. The email, paraphrased, read, "I hate my [device] and I do not have time to learn majordomo, so please remove me from this list." Never mind the fact that to get on the list one has to use majordomo, an automated mailing list handling system, to subscribe. I especially liked this one, again paraphrased, "It would be nice if there were a listname for requests concerning this mailing list, and nicer still if there were an automated system to handle said requests. Please remove me from this list." The mailing list in question had been managed again by majordomo for at least a year.

These are just a slightly more obvious versions of what I see everyday on the 20 or so mailing lists I currently subscribe to. The fact that the lists I am on are mostly used by power-users and system administrators, supposedly the best of the Internet, makes it all the more depressing. The Internet is composed of so many people of so many different backgrounds and occupations, and the medium is so prone to create misunderstanding when used carelessly, that it is imperative that we who are the early explorers, the frontier blazers, get it right, and practice some common courtesy, common sense, and basic thoughtfulness as we ply our trades on this new frontier.

```
To: foo-invite-request
Cc: bigmac@erg.sri.com
Subject: Please unsubscribe me from this
list
```

Thanks. . . and sorry for the rashness of others . . .

System Administration Tools Your Vendor Never Told You About: The Beeper

by Elizabeth Zwicky
<zwicky@corp.sgi.com>

I hear many of you out there shrieking in disbelief. Yes, I wear a beeper. No, I don't hate it. No, I'm not any crazier than I was when I suggested the squeaky octopus (doesn't that make you feel better?) Here are some of the reasons why a beeper works for me:

I have my own beeper. An on-call beeper is cheap for the employer, but is a lousy deal for the employee, because all you get out of it is misery. (This can be somewhat alleviated by making sure you get misery *and money.*) A dedicated beeper is not just a way for your work to annoy you, it's also a way for your friends and loved ones to annoy you. Somehow, most of my pages end up being about food. A modern pager or voice-mail service can allow you to transfer a central number from one pager to another, making multiple pagers also serve the on-call purpose. Of course, like anything good, it will cost more.

If I can't do anything about a page, I make certain I don't notice it. If I'm teaching, for instance, I set it to vibrate and take it off; I can check on it when I can answer it. If you don't have a car phone, and you're in a traffic jam, you don't need to know whether or not your pager is going off.

I don't get paged much. The pager isn't there so that just anybody can get hold of me. In my previous job, we used a simple screening mechanism; everybody had beepers, but the number for the most senior person was the only one published to the world. If it wasn't worth paging him, it obviously wasn't that important. If you paged him anyway, it was not a mistake you made twice.

I have had a pager long enough to have become accustomed to the idea that a significant fraction of the pages are wrong numbers. I don't even try any more; if the page doesn't make sense, I wait and see if I get another one. That weeds out most of the random pages.

I have an alphanumeric pager. It gets email. Most of my pages are real messages in and of themselves, and I don't need to fool around calling people back. (This helps avoid the electronic accretion effect, where you get the cell phone so that you can answer pages, and then the laptop so you can dial in over the cell phone to fix the problem.) Email pages can also be automated: the unscrupulous use this to get out of meetings via "at" jobs; the scrupulous use this to be notified when things run amok.

I have a full appreciation of the magic words "I'm leaving now. If you need me, you can page me." The second sentence is what keeps people from screaming and grabbing your ankles when you say the first sentence. Then you go away and have a life.

Certainly, a pager may not be the right thing for you, but if you find yourself in on Saturday "in case something goes wrong," go for it.

Who Are My Peers?

Elizabeth Zwicky
<zwicky@corp.sgi.com>

System administrators are frequently exhorted to think of the people they support as colleagues, but it doesn't happen often, in either direction: system administrators think of the people they support as users, or "lusers", and other people think of the system administrators as office help or fascists, despite all attempts to encourage other attitudes. (It doesn't help that such attempts usually are aiming for "necessary evil" as an improved attitude.)

The fact is, if you are a Nobel-prize-winning physicist, and you have a brilliant idea about physics, you call a physicist. If you are a Nobel-prize-winning physicist, and you spill your coffee into your keyboard, forget your password, or unplug your computer by mistake, you call a system administrator. The system administrator may have been told that really, as a physicist, you're top notch, but the evidence to hand is going to suggest that you are not really fully in contact with the world around you. This is all very well for a Nobel-prize-winning physicist – nobody expects them to be completely normal – but the same effect holds for lesser mortals like computer programmers, and is likely to be taken as proof that they are not all that bright. If the physicist does happen to call up the system administrator to share the news of his brilliant idea about physics, the system administrator is rarely able to make much sense of it, which doesn't advance the cause of mutual respect, either.

In turn, system administrators are generally apparent to the people as bearers or receivers of bad news. When the IRS calls, your first thought is not "Gee, my refund must be larger than I expected," it's "AUDIT! AUDIT! AUDIT!" When you see mail from a system administrator, your first thought is not "I wonder how my life has been improved now?" it's "What died?" You are also apt to be a little uncomfortable around someone who generally sees you at your worst, particularly when you have a strong feeling that they laugh about it.

More fundamentally, the sense of team membership that people are looking for depends on working together towards common goals. While system administrators and the people they support do have common goals, these tend to be large and abstract; everybody wants the company to succeed, for instance. In day-to-day life, people spend most of their time thinking about and working on smaller, more immediate goals, and those goals rarely involve system administrators and other employees as partners. System administrators generally interact with too many people to be real team members; a system administrator normally supports a number of different teams, and you can't be a sixth of a team member.

So what can you do about it? First, you can try to improve communication in several different directions. Arrange things so you're talking to people when they're not already frustrated and upset. Find out what other people really do, so that you can think of them mentally as "the algorithms expert" instead of "the guy with the sticky keyboard." If people are willing to listen, explain things to them–most of them have no idea about things that we think are intuitive, as evidenced by the person who wanted a five minute explanation of the procedure for installing a device under UNIX. Any device. Under any version of UNIX. He thought there was exactly one procedure, the same for terminals, printers, disks, and so on; after all, devices are files, right? No wonder he thought system administration was easy!

Second, you can redefine the target. You're not going to achieve a situation where the system administrators are team members on every team they support, and you're unlikely to get a ratio where each system administrator supports exactly one team. Stop trying; aim for mutual respect and communication. Look for peers among other system administrators, and work at being a respected and respectful outsider among the people you support. Realize that this doesn't come automatically, and base your attempts to communicate on an understanding that you and the people you support come from different backgrounds and have different sets of expertise and knowledge.

Admittedly, this is widely regarded as terrible advice, and for good reason. "Separate but equal" has never been a highly successful theory. On the other hand, failure to acknowledge reality is not a highly successful theory either, and the reality is that system administrators are more like umpires than team members. It is therefore going to take real work to achieve a good relationship, and that work is not simplified by attempts to claim that system administrators and the people they support are peers, but they're just too dense to notice it.

Celebrating Women in Computing

by Vicki E. Jones
<vjones@cs.uiuc.edu>

Recently I attended the first ever Grace Hopper Celebration of Women in Computing Conference, held June 9-11, 1994 in Washington, D.C. The conference, the vision of Dr. Anita Borg, was described in its brochure as "a technical conference featuring talks by prominent women in computing, panel discussions, workshops, and birds-of-a-feather sessions." Dr. Borg and her program chair, Dr. Telle Whitney, gathered an impressive group of speakers – accomplished leaders from the major technical computing disciplines, representing the academic, government and industrial communities – to present the most recent technological advances in their respective areas of expertise.

The conference's many goals included providing technical exposure to the state-of-the-art about all areas of computing, facilitating collaboration and information exchange in order to work to solve open problems which cross disciplines, recognizing the significant role women have played in the development of computing technologies, and fostering mentorships.

The Celebration is dedicated to the memory of Rear Admiral Grace Murry Hopper. As a member of the Bureau of Ordnance Computation Project at Harvard University's Cruft Laboratories, Admiral Hopper worked on the Mark series of computers and later joined Eckert-Mauchly Computer Corporation where she aided in the design of the first commercial large-scale electronic computer, the UNIVAC I. Subsequent projects allowed Admiral Hopper to refine some of her ideas about broadening the audience of computer users by creating tools which were programmer- and application-friendly.

Admiral Hopper and her team developed several early compilers and participated in the COBOL language design. She returned to the Navy to lead the development of a set of programs and procedures for validating COBOL compilers. Following her retirement from the Navy in 1986, she became a senior consultant to Digital Equipment Corporation where she remained for several years. Admiral Hopper received numerous awards for her contributions to the field of computing.

The Grace Hopper Celebration honored her and a few of the many other women who have contributed to the advances in the computing field over the last twenty years. Admiral Hopper's love for her work and quest for knowledge is shared by all of the speakers at the Celebration.

The keynote speaker at the conference was Anita Jones, Director of Defense Research and Engineering at the Department of Defense. Dr. Jones is one of the few Computer Scientists in the United States with a policy job, and she encouraged other computing professionals to seek opportunities to influence the policy of the United States. Dr. Jones discussed the role of the scientist in providing for the security, prosperity, and health of the US and suggested that scientists should be participating in research that is guided by the goals and priorities of the nation. She outlined some of the principles that guide the Department of Defense in investment in science and technology and how the DoD's strategic research program should be considered a model for future societal support of basic research in the broader national context. The DoD has a balance of goal-directed research to meet specific program needs and basic research to provide long-term advances.

Dates to Remember

8TH SYSTEMS ADMINISTRATION CONFERENCE (LISA VIII)

Conference:	Sept. 19-23

VERY HIGH LEVEL LANGUAGES SYMPOSIUM

Papers due:	Sept. 12
Symposium:	Oct. 26-28

SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION

Authors notified:	August 8
Papers due:	Oct. 4
Symposium:	Nov. 14-18

WORKSHOP ON IEEE MOBILE COMPUTING SYSTEMS AND APPLICATIONS

Abstracts due:	August 20
Authors notified:	Oct. 1
Papers due:	Nov. 15
Workshop:	Dec. 8-9

WINTER '95 TECHNICAL CONFERENCE

Authors notified:	August 31
Papers due:	Nov. 14
Conference:	Jan.16-20, 1995

2nd SYMPOSIUM ON MOBILE & LOCATION-INDEPENDENT COMPUTING

Abstracts due:	January 2, 1995
Authors notified:	January 23
Papers due:	March 6
Conference:	April 10-11

CONFERENCE ON OBJECT-ORIENTED TECHNOLOGIES (COOTS)

Abstracts due:	March 6, 1995
Authors notified:	April 3
Papers due:	May 15
Conference:	June 26-29

Following the keynote there were technical presentations from many disciplines of computing:

- Mary Jane Irwin gave a compelling talk about the MGAP (micrograin array processor) family of architectures she designs at Penn State and Karen Sparck Jones, University of Cambridge, presented her research in linguistically-expressed information.
- Some of the challenges for future compiler research were outlined by Fran Allen, IBM, including portability and performance across systems, issues in parallel and distributed systems, and object-oriented systems.
- Dorothy Denning, Georgetown University, provided information on the encryption technology used by the Escrowed Encryption Standard adopted by the US government and discussed the challenges of an international escrowed encryption scheme.
- Mary Shaw, Carnegie-Mellon University, discussed the codification of software engineering cycles: past success with abstract data types and objects and current endeavors with architectures for software systems.
- Group-enabling software was demonstrated by Irene Greif, Lotus Corporation, who discussed the importance and future of computer-supported cooperative work.
- Maria Klawe, University of British Columbia, talked about her new research goal to understand how interactive electronic media can be used to help children learn mathematics and science.
- Challenges of language implementation support were presented by Susan Graham, University of California, including the development of expressive application-driven languages that are efficiently executable on modern hardware.
- The Thor object-oriented database system was presented by Barbara Liskow, MIT, who discussed its qualities as a good basis for programming distributed applications.
- Elaine Cohen's research, University of Utah, aims to enhance the ability to design, visualize, and manufacture physical parts. She presented a video demonstrating some of the techniques and results of her work.
- Discourse processing and systems for human-computer collaboration are being researched by Barbara Grosz, Harvard University, who presented some results.
- Ruzena Bajcsy, University of Pennsylvania, discussed her work with robots and the challenges of studying intelligent behavior.
- The final talk, by Nancy Leveson, University of Washington, discussed the changes needed in software development to build safety-critical systems.

More information on the conference participants can be accessed on the World Wide Web at URL *http://www.digital.com/pub/doc/hopper/info.html.*

In addition to the technical talks, the banquet speaker, Mildred Dresselhaus, talked about her career in science. There were two panel discussions: "Computers and Policy Issues: They Impact our Lives" and "The Management Option." The policy panel debated currently "hot" issues such as the Escrowed Encryption Initiative (the "Clipper" Chip) and the National Information Infrastructure (NII).

The Management panel discussed management options for women in computer science and engineering.

As a soon-to-graduate graduate student, I found the sessions on cultural and gender issues, on the computing environment for women in Europe, and on careers in academia, government, and industry workshops particularly informative.

Highlights from the conference will serve as the initial content of an archive to encourage young people to include computing among their career possibilities. The archive, officially known as the Computer Science Women's Video Archive, is still in the planning stages and will also include a Women in Computer Science lecture series, a CDROM appropriate for distribution to high schools and junior high schools which will show the more personal side of women which have chosen computing as a profession. For more information about the project contact:

Dr. Anita Borg
Network Systems Laboratory
Digital Equipment Corporation
250 University Avenue
Palo Alto, CA 94301
email: <*borg@pa.dec.com*>

Beneath Who Knows How Many Freudian and Calvinistic Strata

by Jean Pooquet

All of us who have lived through the transitions of the computer industry over the past twenty five or so years are indeed survivors. Improvements in software engineering tools, the advent of "usability" as a product design requirement and the feature improvements in programming languages, among a myriad of other things, have undoubtedly led to a simplification of our lives. But all that's old hat, you've heard it all before.

While the changes in the software aspects of the computer industry have been revolutionary, analogous changes took place in hardware. You remember hardware. Those are the devices, the physical entities, you touch or hear, or sometimes, unfortunately, smell. There has been a continuing miniaturization of computer components that seemingly benefits nearly everyone. The power formerly contained in a three ton, power hungry, room filling behemoth now fits conveniently in one's work area. By the way, I never used to have a work area in a cubicle. I had a desk in an office. I suppose terminology changes, too.

What I want to discuss is a phenomenon observed over the past few years. We've all attended meetings wherein participants have set up, and operated, some small portable computer. You may even recall the earliest of them. They weighed as much as 12 pounds, took up a large amount of table space, had a battery lasting a half hour, and ran operating systems in command line mode. I call them hernia portables. You could carry them but if you did you'd be unable to laugh comfortably for a month or so.

After the so called portable computers came the lap top and hence came the hand held. There is no doubt that this shrinking of physical size and environmental requirements took place with a similarly outstanding increase in functionality. They are getting smaller, lighter, longer lasting, and more powerful to boot (this is the only pun for which I plead for your indulgence).

Consider this typical meeting. I find myself in a meeting where the majority of participants brought and are using their portable computer. As the meeting was about to start, a scrambling began. Nearly everyone wants to sit near a power outlet. I note that some of the more enterprising people brought an extension cord with multiple sockets. The truly polite people permit other participants to plug into their extension cord.

After the meeting starts, the clicking begins. I have a revelation . . . one could determine the relative stressfulness of agenda items by the key-click-induced noise level. The meeting proceeds apace until the first coffee break. That's when the most striking phenomenon sticks out.

People start comparing their portable computers. Memory, processor type, processor speed, disk, display resolution, battery life. Two distant, somewhat musty, memories buzz around in my mind. The first is a class in Introductory Psychology, a lecture on Freud, or was it Jung, or even Calvin? The other is a shower after an eighth grade gym class. My mind operates at a terrible rate. Beads of perspiration trickle down my back. Then comes the revelation . . . I've seen all this before! It's PC envy.

Optimizing Your Shell Scripts

by Scott Hazen Mueller
<scott@zorch.sf-bay.org>

Why Optimize Shell Scripts?

Perl programmers will probably laugh and say the best way to optimize a shell script is to rewrite it in Perl. While I'm not interested in starting a religious war over the use of Perl versus shell, I will say that my personal degree of familiarity with shell makes writing scripts a matter of little more than '`cat >foo.sh`', at least for simple scripts.

Furthermore, I am quite willing to concede that Perl holds the speed advantage over shell for long scripts or for long processing jobs, where the 6x load time (0.3 secs against 0.05 secs to run an empty script [ISI 68K20, BSD 4.3 UNIX]) is amortized over a much longer run time. However, for small jobs the shell is still quite handy; it's easy to prototype in, the tools are diverse if not exactly orthogonal, and for complicated processing sed and awk (and Perl!) are available to handle the tricky parts. In one recent instance I cut the run time for a shell script down from about 7.5 seconds to under 2.5, a 3x improvement, without having to code it into a new language. As system administrators, we write and use a lot of script tools. Cutting the run time of 'legacy' scripts down 2 or 3 times may not seem like a big deal, but having something beat on your workstation disk, and make all your windows swap out, for 30 seconds instead of 90 could be a nice win.

I will write entirely about Bourne shell; I don't use C Shell for programming and don't have Korn shell available where I do the bulk of my work.

Basics of Optimization

Code optimization is a well-understood part of scientific computing. Some basic techniques include loop-unrolling; moving computationally expensive work outside of loops and pre-calculating data and caching it; keeping data in registers instead of in main memory; and avoiding extra I/O when possible.

Some of these techniques need not be applied to shell scripts. For example, unrolling a simple loop is neither a win nor a lose in shell. The following program:

```
#!/bin/sh
for i in 1 2 3 4 5 6 7 8 9 10
  do
  ls >/dev/null
done
```

executes in the same time as the equivalent unrolled loop:

```
#!/bin/sh
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
ls >/dev/null
```

Likewise, I/O operations seldom dominate the execution time of a shell script, with the exception of the I/Os associated with `fork(2)/exec(2)`. Even if they do, there's not much that can be done about it, unless the script makes excessive use of temp files. If your script needs to process partially-processed data twice, for example, the data will need to be written to a temp file, but the cost of doing that can often be reduced by using `tee(1)` so that the data need only be read from the temp file once.

Pre-calculating Data

In a scientific application that loops over multiple variables, it makes sense to perform some calculations once per outer loop instead of on every inner loop. For example, if the program is calculating a function of x, y, and z, the first approach might be the following:

```
for ( x=0; x<X_MAX; x++ )
   for ( y=0; y<Y_MAX; y++ )
     for ( z=0; z<Z_MAX; z++ )
       funcval = x * y * z;
```

This code will perform 2 multiplications for every iteration of the innermost loop. If X_MAX, Y_MAX, and Z_MAX are 1000, then the program will perform 2 billion multiplication operations. Consider, instead, this code:

```
for ( x=0; x<X_MAX; x++ )
   for ( y=0; y<Y_MAX; y++ ) {
     temp = x * y;
     for ( z=0; z<Z_MAX; z++ )
      funcval = temp * z;
     }
```

It may not look very different, but the new code performs 1 billion plus 1 million multiplications, just over half as many as the original version.

Now, few shell scripts perform heavy computation, but similar operations can be performed, pre-calculating and caching data. Consider a program that displays a menu of options in a free-text format, and uses a stripped-down form of the option name as the return value for a data search. For example, a skills checkoff list might contain:

C Programming
HTTP/WWW (Mosaic)
TCP/IP Networking

It's rather difficult to use these values in regular expressions, so we might want to strip whitespace and special characters. The following code fragment generates an HTML <SELECT> menu:

```
while read item
   do
   echo <option value=`echo $item|sed -e /
     "s/[ /()]//g"`> $item
done <itemlist
```

The successive values of the 'value' keyword would be:

CProgramming
HTTPWWWMosaic
TCPIPNetwork

This is an expensive piece of code because it calls `sed` on every iteration. It would be better performance-wise to move the `sed` command outside the loop. However, it is not possible to just pipe the `sed` output into the loop, because inside the loop we want both the original free-text form as well as the stripped form.

A significant performance enhancement is to take the original input and generate a cache file:

```
sed -e "s/[ /()]//g" < itemlist > $tm1
sed -e "s/ /-/g" < itemlist > $tm2
pr -m -t -s' ' -w80 $tm1 $tm2 | sed -e /
    "/^ $/d"  | sort > cache
rm -f $tm1 $tm2
```

This may not seem like a win, but the cache is built only a few times compared to the number of times the menu is built, and for a longer item list the time that would be spent calling sed inside the loop dominates the time spent rebuilding the cache.

The cache can then be read in and processed with shell built-ins in a manner similar to the method discussed below. The optimizations used here were part of those done to go from 7.5 seconds to <2.5 seconds as mentioned previously.

Fork/exec Considered Harmful

One of the biggest gains in shell script performance can come from avoiding `fork()` and `exec()` calls. The shell makes a `fork()/exec()` call every time it needs to load a program. Therefore, by not loading programs, these calls can be avoided. Basically, you use the shell's built-in functionality rather than calling an external program. This is, as it happens, the main win Perl has over the shell, because almost all of the functionality needed to write useful programs is built in to Perl. However, it is still possible to accomplish useful work in shell using just the built-ins.

The most overlooked shell built-in is the '`set`' command. It is possible to use '`set`' to break input into digestible chunks, and in many cases to avoid the use of `sed(1)` or `awk(1)` for input processing. In conjunction with '`IFS`', it's possibly to process almost any regularly-formatted input.

Consider, for example the password file. The following shell script will extract the GCOS field entirely with built-ins:

```
while read pwdline
 do
 if test ! -z "$pwdline"; then
  IFS=: set $pwdline
  gcos=$5
 fi
 echo $gcos
done
```

Contrast this with the traditional method:

```
while read pwdline
 do
 gcos=`echo $pwdline | awk -F: '{print  $5}'`
 echo $gcos
done
```

(The scripts above are coded as if the `$gcos` variable will be used within another part of the loop.)

The run times for the two forms are 1.2 seconds for the first script against 3.0 seconds for the second. The moral? Avoid making unneeded program calls inside loops.

This method uses three separate shell programming tricks. The first is that an environment variable can be reset for the duration of a single command by using the syntax 'VAR=value command'. This saves the entire sequence

```
oldVAR=$VAR
VAR=value
command
VAR=oldVAR
```

Second, `IFS` is a special value, the shell's Internal File Separator. `IFS` defines the set of characters that separate fields of the shell's input. An important thing to know about `IFS` is that it also separates the fields of the shell's argument vector string. This is important because the '`set`' command resets the argument vector to the value given as the '`set`' parameter. That is, if a script is called with

```
sh ./foo.sh arg1 arg2 arg3:arg4
```

then $1="arg1", $2="arg2" and $3="arg3:arg4". If we use the command

```
IFS=: set $3
```

then the old argument vector is replaced, and $1 becomes "arg3" and $2 becomes "arg4". Obviously, you need to preserve the original arguments before using 'set' to reset the argument vector. (Note: this trick may not work on SunOS 4.1.x's `/bin/sh`, at least for IFS. It does work on the shell I originally wrote the sample script on.)

Another point to note, one that doesn't really qualify as a trick, is that most shells provide '`test`' as a built-in. If '`test`' is a built-in, then it is fairly fast. If not, then it will cost a `fork/exec` for every conditional. The really important thing to know here is that the syntax

```
if [ condition ]; then
   action
fi
```

will always cost a `fork/exec`, because '`[`' is not a shell built-in. Take a look in `/bin` or `/usr/bin` for a program called '`[`'; it should be there and should be a link to the `test(1)` binary.

Use Filters as Filters

A last, related performance point is to avoid calling programs such as `awk` or `sed` on each iteration of a loop. These programs are designed as input filters and should be

used as such. For example, a simple shell loop might perform some primitive comment processing:

```
while read line
   do
   set $line
   if test "$1" != "#"
     echo processing $line
   fi
done
```

The input file this script uses is limited in that comments can only start on the first column of each line, and the '#' comment character must be separated by whitespace from the next character. A more sophisticated attempt at comment processing might look like:

```
while read line
   do
   line=`echo $line | sed -e "s/#.*//"`
   if test ! -z "$line"; then
     echo processing $line
   fi
done
```

This, however, is getting expensive again, calling `sed` once for each and every line of input. A much better method is to use `sed` in a pipeline:

```
sed -e "s/#.*//" -e "/^$/d" | while
              readline
   do
   echo processing $line
done
```

Not only does this pull the `sed` call out of the main processing loop, it also eliminates the associated test, because we now can guarantee that the main loop will never see a comment. Furthermore, we can use sed to strip out empty lines, guaranteeing that the main loop won't see them either.

Conclusion

If your shell scripts take too long to run, or use too many system resources, a few simple programming techniques can speed them up considerably. Make sure to take advantage of the filter nature of basic UNIX utilities such as `sed` and `awk`, and use as few operations inside your main loop as possible. In conjunction with careful use of shell built-ins, shell scripts can be made to run 2-3 times faster in just a short time.

Conquering Corporate Computing with Message-Oriented Middleware

by Tim Daneliuk, Consulting Engineer, Covia Technologies
<tundra@ct.covia.com>

Introduction

In the past several years, we've begun to see new technologies make significant inroads into corporate computing. What was once the sole domain of IBM mainframes is now becoming a multi-vendor battleground littered with a variety of operating systems, hardware, and networks. Its worth noting that this does not mean that corporations are converging exclusively on UNIX and other so-called "open" technologies. In fact, the typical large computing enterprise most resembles a computing supermarket with at least "one of everything."

For the software and systems vendor selling to large corporations is very attractive. These enterprises have, relatively speaking, the biggest checkbooks. For a small software or systems developer a single corporate sale can be the defining moment which establishes a stable and growing business. To get there though, you have to know how to conquer the Tower Of Babel of technology such environments present. In this article, I'll briefly introduce the use of message-oriented "middleware" as a means to unifying heterogeneous environments. I'll also discuss why this approach is preferable to the existing models for interoperability such as Remote Procedure Calls.

Honesty compels me to confess that I work for a firm which produces such middleware (Communications Integrator from Covia Technologies). However, the intent here is to inform, not sell, so the material which follows is fairly generic.

What Drives Corporate Technology?

Before we consider the technical issues, it's worth spending a little time trying to understand the forces which drive technology decisions in large organizations.

The first, and single most dominating factor, is economics. For instance, contrary to the folk myths found in our industry's popular press, there is no overwhelming movement in Corporate America to abandon existing technologies and converge solely to UNIX and TCP/IP. In fact, this may well

never happen. The reason is simple: money. Large corporations, especially, have a staggering investment in their existing infrastructure and applications. Even if they wanted to, it is simply not financially realistic for them to make the switch to a single, monolithic computing model. So, despite what you may hear and read, don't expect to see a lot of unplugged mainframes in the alley anytime soon.

A second, closely related, driver is that *applications* are the "cash register" of any company. Preserving existing applications, writing new ones, and maintaining them all are the real core focus of a corporate IS shop. Things like networking, operating systems, and even hardware are largely viewed as necessary evils and are virtually never seen as an end unto themselves.

The consequence of all this is that no new technology will be taken seriously if it cannot embrace the existing "legacy" applications. Without question, the most important of these legacies live on mainframes running MVS possibly with CICS or IMS transaction processing monitors. It is also overwhelmingly the case that such systems are networked using some flavor of SNA.

In this legacy case, "embrace" means more than just being able to connect and communicate with such applications. It means being able to truly interoperate. In the client-server world we think of using distributed computing to move *parts* of an application into various classes of computers. In legacy interoperability the trick is to make *whole* applications which stand-alone work together with new technologies in some coordinated fashion. When you consider the fact that these applications run on vastly differing platforms that do not even share a common networking fabric, you begin to see the incredible integration complexity the enterprise computing designer faces. At a minimum, such integration has to accommodate technology from IBM, Microsoft, Novell, and, of course, a variety of UNIX flavors.

The final corporate technology driver is risk management. At today's pace of innovation, a primary infrastructure technology (like hardware, operating system, or network) goes through another "generation" every 2 to 3 years. By contrast, mission-critical enterprise applications can live for 15 or more years. So, in the average lifetime of an application, there can be as many as 7 infrastructure "roll overs," upgrades, or changes. It is crucial that these applications, which are the lifeblood of the business, be protected and preserved as the infrastructure beneath changes. Large computing enterprises are thus increasingly selecting technologies which "insulate" their applications from this constant threat of change.

What Is Middleware?

"Middleware" has become a much-abused term in the computing press lately. It seems to have taken the meaning of being any applications programming interface (API) not part of the operating system proper. While this is perhaps broadly correct, true middleware has a number of attributes which distinguish it from other types of software:

- A primary focus of middleware is to provide *coordination of distributed elements* in the enterprise. The idea is to make the many disjointed pieces of a large system operate as a unified whole.

- Middleware serves the *applications programmer* by providing simplified programming syntax and semantics for invoking the underlying infrastructural services. The idea is to provide the applications programmer with a way to quickly exploit traditionally complex services like networking without having to be familiar with all the low-level details.

- The abstraction mechanisms of middleware strive to present a *generic view* of the underlying service. For instance, properly implemented networking middleware "hides" the specific details of the local communications protocols. The applications programmer sees the enterprise network as being one logical address space with common (simple) communications semantics even though there may be many different protocol stacks in actual use.

- The best middleware is implemented in *many heterogeneous* environments. By its nature, this kind of software is designed to conquer enterprise-wide heterogeneity. It must therefore be available for the dominant mainframe, mid-range, and desktop platforms.

- Middleware is generally *implemented in user space*, not as a kernel enhancement or device driver. By acting as a well-behaved application itself, middleware is minimally exposed to the risk of kernel and internals changes as operating systems are patched and upgraded.

Notice that many of the current distributed processing components, even in the "open" systems community, do not realize these goals. For example, one could argue that TCP/IP conquers networking heterogeneity at some level. It is certainly true that TCP/IP might be used to connect the wide variety of systems found in the large enterprise. The problem, though, is that this is a kind of virtual *homogeneity* in which every networked node is forced to converge on a single protocol. As we saw previously, this kind of replacement strategy is typically too expensive to consider

seriously in a large environment. Moreover, TCP/IP does not yet necessarily run well in all environments. ("Well" in this case means that the protocol be reliable, highly-available, efficient, and manageable.) It is still pretty much the case that mainframes run SNA best and TCP/IP runs best on UNIX or desktop systems. For these reasons, the abstraction mechanisms necessary to unite heterogeneous systems must live well above the protocol level. This has led to the development and growth of middleware.

What is Message-Oriented Middleware?

One of the basic issues faced by the middleware designer is deciding what semantics or operational paradigm to present to the applications developer. Many such paradigms have been proposed including Remote Procedure Calls, Distributed Queues, Distributed Filesystems, and Distributed Operating System Kernels. Each of these models has enjoyed vigorous research attention as well as varying degrees of commercial success. To date, however, they have largely not found much use in the traditional mainframe and legacy applications domain.

When the constraints of legacy system integration are considered, one of the most successful paradigms has been that of "message passing." In part, this is because many legacy applications were written under the umbrella of a Transaction Processing (TP) monitor such as CICS. TP monitor-based applications typically are designed to interact with a population of remote terminals such as IBM's 3270 family. As such, the user keys in a request for work, presses the transmit key, and waits for their screen to be painted with a response from the application. This itself is essentially a message passing paradigm. The user sends a "message" to the TP monitor which dispatches the appropriate application code. When the computation completes, a screen "message" is returned. It is thus fairly straightforward to take existing legacy applications and "hook" their points of ingress and egress so that they can accept a more generic source of messages (i.e., a message generated by something other than a terminal).

Message passing is also a powerful notion when integrating diverse stand-alone applications. The idea is strongly object-flavored. By creating a public messaging interface to each application, these stand-alone legacies become network-attached abstract data types in some sense. That is, the public interface does not reflect the underlying implementation, but merely provides a published way to invoke work. The applications designer coordinates the work of new and legacy applications by defining the appropriate content and message order which must flow between them.

Because messaging maps well into both legacy and emerging technologies, so-called "Message-Oriented" middleware is becoming increasingly popular. This sort of software has all of the attributes of middleware described above implemented around a distributed message passing model. Typically, message-oriented middleware will have a number of key features:

The exchange of variable length messages provides a *unified representation* for units of work across all supported platforms, operating systems, and networks.

- The middleware provides *position-independent* messaging. Whole applications or fragments of applications each are addressable entities in one logical address space. This address space does not have to be correlated one-to-one with the physical topology of the networks involved. The most important implication here is that the logical distributed application topology need not look anything like the underlying network. For instance, even if the networks are hierarchical with a central computing site, distributed peer applications can be effectively implemented.

- Messages can be sent in a *time-independent* fashion. That is, they can be sent reliably to presently unavailable destinations. In this regard, a messaging system can resemble a distributed queuing manager. The middleware accepts the message and attempts continuous delivery until the destination does become available. This feature is of enormous value in large systems where there may be literally tens-of-thousands of computers involved. It is bluntly impossible to build reliable applications at this scale if every node involved must be on-line for the application to proceed. This feature allows the consumers and producers of data or services to operate at different moments in time.

- The middleware itself acts as a *logical transport* across many different protocols. As such, the middleware does not rewrite the contents of the message for any purpose such as data presentation. For instance, data encryption or code set conversion are presumed to be the responsibility of the code above the middleware. If such work must be done frequently, it can be written as a stand-alone distributed service and thus become middleware-addressable.

- Messages may be initiated with either *synchronous* (spin-lock) or *asynchronous* (immediate return) semantics. This is possibly the single most important feature of a well-developed messaging system. Asynchronous messaging allows the designer to post messages into the enterprise without waiting for each response serially. This maximizes total work throughput by exploiting the underlying parallelism inherent in having many stand-alone applications running simultaneously.

- The application may select the level of *message integrity* desired on a per-message basis. It is well understood that a high level of integrity across a network requires a correspondingly high level of bandwidth and system resources. These are needed for session management, retry logic, flow control, and so on. In many cases, this is not necessary and would be overkill for many kinds of work. For instance, a two-phase commit would almost certainly flow via highly-assured messages. However, updating the clock display at a bus station every 30 seconds could easily be done with logical "datagram" messages. If one message failed to arrive, the next would correctly update the clock.

- The middleware provides the *message association* between messages sent and received. This is particularly important in the case of asynchronous semantics where message response is investigated long after the initial message has been sent.

- A *priority system* of some sort is used to define inter-message sequencing. A given piece of middleware typically serves many applications simultaneously. By associating a priority value with each message, the middleware can expedite the delivery of the most critical messages.

Because of these attributes, message-oriented middleware provides a powerful mechanism for integrating highly heterogeneous systems on a very large scale while preserving system availability and integrity.

What about RPCs?

One of the most-often suggested schemes for distributed processing has been the Remote Procedure Call. Certainly the UNIX community has several popular examples of this technology in use. Space does not permit a detailed analysis of why RPCs have failed to conquer the large enterprise, but a number of ideas are worth mentioning in passing:

- RPCs are typically single-protocol. Even so-called "transport independent" RPC implementations can engage only one protocol at a time. For all practical purposes, RPCs must be considered a TCP/IP-only technology. This makes them of limited value in the multi-protocol corporation.

- RPC implementations are not widely available outside the UNIX and desktop computing community. Even if the enterprise were willing to converge on TCP/IP as the exclusive protocol, the question remains whether this distributed processing model could be made to run effectively in traditional mainframe environments.

- The RPC paradigm does not map well onto legacy applications. As described above, legacy applications naturally map well onto the messaging model. The function call-return mechanics of RPC have thus far failed to serve this sort of integration task well.

- RPCs are mainly designed to operate as synchronous services. The client code does a function call and then waits for a response. This is extremely inefficient, particularly if the node doing the call is a server of some sort. In effect, the server has to serialize its work waiting for responses. Certainly this can be ameliorated somewhat by threading, but such facilities do not exist in all relevant environments. In some cases, the RPC vendors have made provision for non-blocking or asynchronous RPCs. However, this has been at the expense of reliability. One gets non-blocking RPC over UDP which places the responsibility of end-to-end integrity on the shoulders of the application.

- The RPC function call model expects the caller and called code fragments to be simultaneously available. As described above, large-scale systems availability is compromised if every computational node must be simultaneously available for an application to proceed.

- RPCs have no mechanism for network-wide call optimization. When integrating the work of many stand-alone applications, an initial request for work may flow among many different application servers before the task is complete. With messaging, the last server invoked can return directly to the initial requester simply by sending one last asynchronous message. By contrast, if each step of the computation involved an RPC, each call must return to its caller, thereby doubling path length through the network.

- The two commercially dominant RPC mechanisms (Sun ONC and OSF DCE) cannot interoperate. This is ironic given the loud emphasis these organizations place on "open" computing.

Summary

Middleware is becoming a popular method of alleviating the workload of the applications programmer. Large corporate environments are increasingly implementing message-oriented middleware as a means to preserving their legacies, embracing emerging technologies, and integrating systems in the large. As we see UNIX and its derivative technologies gain a greater foothold in the corporate boardroom, we will, no doubt, end up using middleware in the computer room.

What's New

EDUPAGE
<edupage@ifovy.educom.com>

Edupage, a summary of news items on information technology, is provided three times each week as a service by EDUCOM – a consortium of leading colleges and universities seeking to transform education through the use of information technology.

Small Change on the Net. Researchers at Carnegie Mellon University are developing NetBill, a computerized system for tracking and billing users for small transactions, such as a ten-cent charge per document. The developers hope NetBill will evolve into a universal accounting system on the Internet. (*Chronicle of Higher Education* 4/20/93 A31)

Ntia Will Fund Information Highway. The head of the National Telecommunications and Information Administration says its budget will go primarily toward jump-starting the creation of an information highway. $100 million of a $134 (sic?) budget request will go to grant programs. (*BNA Daily Report for Executives* 4/15/94 A32)

Ads (and Flames) on the Net. After sending an unsolicited ad for his legal services to more than 9,000 Internet Usenet groups, a Phoenix lawyer got 30,000 replies, including thousands of "flames" [outraged messages] from persons who objected to his use of the Internet for unsolicited direct mail. Internet Direct, the lawyer's service provider, rescinded the lawyer's account. The lawyer's threatening a $250,000 lawsuit against Internet Direct and is planning to write a book about advertising on the Internet. (*New York Times* 4/19/94 C1)

Email Eavesdropping. One in five companies admits that it eavesdrops on its employees by searching computer files, voice mail or email, but a spate of lawsuits is beginning to curb the habit. If a company plans on monitoring employees, it should tell them in advance to avoid legal trouble later. (*Investor's Business Daily* 4/19/94 A4)

GTE Jumps the Guno n Personal Communications Services. GTE Will Become the first major cellular carrier to offer nationwide pocket-phone service. The fee is expected to be $25 a month, plus a per-minute charge. (*Wall Street Journal* 4/19/94 A3)

Digital Printing Versus Offset Printing. Xerox is announcing software to link computers and scanners to high-speed copiers, regardless of location and regardless of manufacturer. (*New York Times* 4/19/94 C3)

Virtual Exclusion. Some of the people who pay $19.95 a month to participate in Echo, a New York City-based electronic bulletin board (or "electronic salon," as its owner calls it) focused on art, politics, and New York life, are upset a self-selected group of Echo subscribers have been secretly participating in a private late-night conference (or "nightclub") called Xenophobia. "The very existence of an 'in crowd' is in direct contradiction to everything Echo allegedly stands for," one subscriber complained. But Echo's founder and owner says, "People who join the electronic world expecting it to be better than the physical world will be disappointed. And if you expect everyone is going to be your friend, you'll be disappointed, too. I'm sorry. Echo is not Oz." (*The New Yorker* 4/25/94 p.40)

Keeping on Track. Computer trackballs are easier on your arms and wrists than a mouse, and also provide more precise cursor control. (*Bottom Line Personal* 4/15/94 p.5)

64-Bit Chips. If memory prices continue to fall exponentially, multigigabyte TVs will become feasible and will add to the demand for 64-bit microprocessors, such as Digital Equipment Corporation's Alpha 21064. A 32-bit chip, such as the current versions of the Pentium or the PowerPC, can address only 45 seconds of uncompressed video in memory; a 64-bit chip could handle 50 million hours. *(Ed Note: So there you have it.)* (*Forbes* 4/25/94 p.162)

Fax Facts. A Gallup poll has found that fax transmission account for 36 percent of telephone bills at Fortune 500 companies, and that email has made little headway in being used routinely for corporate communications, in spite of its considerable cost advantage. (*New York Times* 4/19/94 C3)

Not Playing Games. Consumer devices that Nintendo, Sony and Sega are developing will have computer power far in excess of what's now available in personal computers; for example, Nintendo's $250 Project Reality video-game player and personal communications device will be more powerful than the Cray I supercomputer of 1976 (*Ed note: as are Pentiums except for floating point*). Nintendo's CEO asks: "If you put something this powerful and this cheap in the market, what happens to everybody else?" (*New York Times* 4/20/94 C1)

Energizing the Battery Market. Three new technologies are emerging to solve the biggest problem laptop users face–short-lived batteries. Lithium ion batteries are already included in Toshiba's newest subnotebook computer, the Portege, giving it a 20% to 30% longer performance than nickel metal hybride-equipped machines. Under development are lithium-polymer batteries which are flexible and can be wrapped around other components, and zinc-air batteries which promise to be lighter and more powerful than any other system. (*Investor's Business Daily* 4/20/94 A4)

CD-ROM Update. The Multimedia Business Report predicts that the number of PCs in the country with CD-ROM drives will nearly double to 9.5 million this year. (*American Journalism Review* 4/94 p.15) CD-ROM prices should drop 40% to 60% by year's end, due to stiff competition and tight shelf space. Of the 1,200-1,500 titles anticipated to be available, only 10% will be stocked by retailers. (*Investor's Business Daily* 4/21/94 A4)

PC Magazine for Kids. *Child Magazine* and *PC World* are teaming up to edit and publish a 32-page section for young computer fans. "Smart Computing for Kids" will appear four times a year in both magazines, starting in September. (*Miami Herald* 4/20/94 C3)

Fiber Optic to Africa. AT&T hopes to use a $1-1.5 billion grid of undersea fiber optic cables for communications among African countries and between Africa and the rest of the world. "African nations need to be connected to the global marketplace," says an AT&T executive. The network would be owned and managed by Africans. (*New York Times* 4/26/94 C4)

Sharing Windows. Microsoft has delivered beta versions of its new Windows program, code-named "Chicago," to about 8,000 software makers, encouraging them to develop applications to work with the new system. Barred from access to Chicago are all software development employees working on a technique called OpenDoc that rivals Microsoft's approach toward linking disparate documents. (*Wall Street Journal* 4/25/94 B6)

Email Messages Released. The University of Michigan has released copies of messages exchanged during a computer conference of the school's regents. The action was in response to requests from two newspapers, which claimed that messages passed among publicly elected officials are public information. (*Chronicle of Higher Education* 4/27/94 A26)

Archeologist Wins Internet Defamation Suit. An archeologist, formerly at the University of Western Australia, has won a lawsuit filed in Australia against an anthropologist, claiming comments made about him on an Internet bulletin board were defamatory. Damages equal to $28,000 were awarded after a psychiatrist testified to the plaintiff's anxiety and depression suffering caused by the remarks. (*Chronicle of Higher Education* 4/27/94 A30)

Racist Message Protested. A 5-page racist message distributed over the Internet set off a deluge of protests to the University of Michigan, where the message originated. The student initially implicated was exonerated when university officials found the computer he had used was equipped with a program that collected passwords. University administrators are still trying to identify the perpetrator. (*Chronicle of Higher Education* 4/27/94 A26)

Government Probes "Pretty Good Privacy" Export. The creator of the "Pretty Good Privacy" encryption program is being investigated by the FBI, and a federal grand jury is examining whether he broke laws against exporting encryption codes. At the heart of the issue is whether Philip Zimmermann was responsible for placing the program on the Internet, where it was picked up and used by networkers in other countries. Zimmermann has claimed he did not personally post any messages containing the program. (*Wall Street Journal* 4/28/94 A1)

TCP/IP Stats. An analyst at International Data Corp. predicts, "By the end of 1992, 6.7% of all PCs on local area networks were using TCP/IP, but by the end of 1993 that was up to 9.4%. By the end of 1994, we estimate it will be over 14%." (*Investor's Business Daily* 4/27/94 A6)

Internet Statistics: The Net Keeps Growing and Growing. Traffic on the NSF backbone grew by a stunning 20.7 percent – nearly 2 terabytes – during the month of March – the largest single jump in the history of the Internet. Gopher traffic grew by 17.6 percent and http (WWW) grew by 32.9 percent to a new total of one-half terabyte per month. Http traffic grew by a total of 0.7 percent of total NSFNet traffic (*Ed Note: Which does not include the ftp requests generated by Mosaic*). (*Internet Society*)

Commodore Calls It Quits. Commodore, a PC industry pioneer, is going out of business and liquidating its assets for the benefit of its creditors. (*Atlanta Journal-Constitution* 4/30/94 B3)

Systems Failure Delays Computer-controlled Airport. An executive for the company that designed a computerized baggage system for Denver's much-delayed new airport says that the latest system test "certainly wasn't a success. A failure? Yeah, maybe. Without declaring defeat we need to be intellectually honest about what we saw." The system is controlled by computers that route baggage to different destinations based on readings from bar codes on the luggage tags (*Atlanta Journal-Constitution* 4/30/94 B3)

Apple Speaks Out for Speech. "Regarding voice recognition," says Apple CEO Michael Spindler, "we've made it a key strategic goal to reshape and enhance the value of personal computing by adding communications features as a standard feature across our entire product line." (*Educom Review* May/June 1994 p.8)

Intel Predicts Price Cuts. Intel is leaking word to its customers regarding further price cuts in the second half of this year. It's anticipated that the cost of an entry-level Pentium chip will drop from $675 currently to $370 next fall. (*Wall Street Journal* 5/3/94 B5)

Intel Wants to Be the Pc-Everything. Intel is betting its future on consumer electronics. "What I'm after," says CEO Andy Grove, "is televisions and telephones and every single-purpose appliance. The best way for us to go for the 250-million-unit market is to move video telephony and conferencing and entertainment and information on to the PC and render those other things less and less relevant." (*Fortune* 5/16/94 p.62)

More RSI Lawsuits Filed. More than 2,000 lawsuits have been filed against big names like Eastman Kodak, IBM and AT&T, alleging the keyboard manufacturers knew of the risks of repetitive stress injuries as long as 10 years ago, and failed to warn customers of possible problems related to prolonged keyboard use. Documents show some companies became aware of the potential for injury through claims by their own employees, and issued internal memos on minimizing injuries, but these warnings were not publicized outside the companies. (*Wall Street Journal* 5/4/94 B1)

Short 'n' Sweet Phone Numbers. Southern Bell is offering a limited number of three-digit phone numbers for sale at $25,000 apiece. Due to high demand, in the next two years, BellSouth will offer a larger number of four-digit numbers to local customers. (*Miami Herald* 5/5/94 A1)

Chasing Mice Away . . . Apple PowerBook will be introducing a 1.8 ounce touch-sensitive pad called GlidePoint which functions like a mouse or trackball but which has no moving parts; users move their fingers across a plastic surface, and tap their fingers on the surface to start programs. (*Atlanta Journal-Constitution* 5/5/94 F2)

Here Comes the Sun. Sun Microsystems is positioning itself as a mainframe-killer, as it attempts to pump up sales of servers, which currently account for just 20% of the company's revenues. Servers are now the fastest growing part of the computer market as companies move huge marketing and financial databases off of outmoded mainframes. "We're going after the database market in a big way," says CEO Scott McNealy. (*Wall Street Journal* 5/10/94 B4)

Lawyers Capitalizeon Internet Flaming. Canter & Siegel, a husband-and-wife law firm that found itself scorched by flame mail last month for advertising on the Internet, has decided to launch a new service, Cybersell, to help other businesses do the same thing. Cybersell will charge $500 for access to 6,000 news groups. "People like Canter & Siegel are taking grotesque advantage of liberating technology that supports the free and open exchange of ideas," says the president of the Internet Company. Retorts Siegel, "Our fate has been that we're making a lot of money. If a bunch of hysterics want to scream and yell and make fools of themselves, then I don't feel they warrant respect." (*Wall Street Journal* 5/9/94 B2)

Internet Top Ten Best-selling Books. CPU Publishing Update says the top ten best-selling books on the Internet are: 1. *Whole Internet User's Guide* (Krol); 2. *The Internet Complete Reference* (Hahn & Stout); 3. *Internet for Dummies* (Levine); 4. *Mac Internet Tour Guide* (Ventana); 5. *Complete Internet Directory* (Braun); 6. *The Internet Companion Plus* (LaQuey); 7. *Internet Starter Kit for the Mac* (Hayden); 8. *The Internet Navigator* (Glister); 9. *Connecting to the Internet* (Estrada); 10. *Zen and the Art of the Internet* (Kehoe). (*Internet Business Report* May 94 p.5)

Compact Disk with Ten Times the Storage. IBM hasdemonstrated an optical disk technology that could produce CDs holding about 6.5 billion bytes of data, by storing information on up to 10 layers on the disk, with each layer readable by having the laser beam individually focused. If the new multilayered technology is combined with the so-called "blue laser" technology (which can double capacity by shortening the wavelength of the laser beam's light to allow a finer focus), it might be possible to make a palm-sized CD that could store the equivalent of several thousand 200-page books. (*New York Times* 5/13/94 C4)

Copy Machine Turns Pages of a Book. Ricoh has developed the prototype of a copy machine with a device that uses static electricity to automatically turn the pages of the book or magazine being copied. You will no longer have to stand at the machine, wetting your fingers, unless you want to. (*New York Times* 5/14/94 p.19)

Another Software Patent to Be Re-examined. The U.S. Patent and Trademark Office will re-examine a broad patent granted two years ago to Software Advertising Corp. The patent appears to cover the way an advertisement can be blended into a computer program and displayed on the screen. By law the Office's commissioner can order a re-examination any time he thinks the issue of "prior art" could potentially reverse a patent claim. (*Wall Street Journal* 5/17/94 B5)

Reach Out and Play with Someone. A system developed by AT&T and Sega Genesis will allow players to link two Genesis game systems over a phone line and compete, aided by an AT&T voicespan technology that will let them speak to each other while they play. (*Atlanta Journal-Constitution* 5/19/94 C10)

Password Primer. The Computer Law Association has some advice on choosing passwords. Don't choose any word in the dictionary, or one that relates to you personally (i.e., birth dates, license plate numbers, etc.). Do use a password with at least eight characters that has a mix of numbers, upper- and lower-case letters, and if possible special characters such as &$@*. (*Bottom Line Personal* 6/1/94 p.8)

Fonts for 260 Languages. Linguist's Software is a ten-year-old company in Edmonds, Washington, that found its niche by developing fonts to allow Macintosh users to compose in the character sets of any of 260 languages, including 15 American Indian languages, 28 African languages, and languages spoken throughout Asia. (*New York Times* 5/22/94 Sec.3, p.9)

IBM Expands Power Pc Line, Sells at Radio Shack. IBM's got a new series of workstations that use the Powerpc chip,\ upping the number of RS/6000 models running on the new microprocessor to eight. (*Wall Street Journal* 5/24/94 B7) Meanwhile, Radio Shack announced a pilot program to sell brand-name IBM computers in about 500 of its stores. (*Investor's Business Daily* 5/24/94 A5)

Attention Librarians–Fatal Bookshelves. A union claims that Library of Congress employees are exposed to potentially fatal crushing hazards caused by sliding mechanical bookshelves, and asserts in a lawsuit that the bookshelves have occasionally started up unexpectedly, undeterred by safety sensors. The government insists the bookshelves have experienced "only three or four failures," none of which resulted in employee injury. (*BNA Occupational Safety & Health Reporter* 5/18/94 p.1787)

Campuses Wrestle with Computer Crime And Punishment. Campuses are increasingly dealing with computer-originated crime and mischief – everything from bomb and death threats, to racist and sexual harassment, to forged messages such as one at Dartmouth that caused students to believe an important exam was canceled. While some students have called for a detailed list of computer commandments, campus officials are wary of limiting their disciplinary discretion. "If you make a long list of dos and don'ts, you are cajoling students into thinking, `How can I squeeze through that?' You should be making them think, `Should I be doing that?'" says a computer science professor at Kansas State. (*Chronicle of Higher Education* 5/25/94 A19)

Newton Not Yet Ready for the Masses. The marketing director of the division responsible for Apple's Newton says the Newton is years away from being a product for the masses. "We think it could become a consumer appliance, but not in the short term. Typically these things take years to happen." (*Atlanta Journal-Constitution* 5/25/94 F8)

Cyberporn is Prosecuted. In two recent cases in Oklahoma and Texas, courts have convicted defendants for using electronic bulletin boards to distribute obscene material. In the Oklahoma case, defense attorneys argued that state obscenity laws don't apply to electronic devices such as CD-ROMs, claiming that what was on the disks was actually binary code. In the Texas case, U.S. Secret Service agents seized computers and electronic equipment from an electronic publisher. (*Wall Street Journal* 5/27/94 B3)

Man Charged in Electronic Stalking. A Michigan man has been charged with breaking a state anti-stalking law for continuing to send email to a woman after she and the police told him to stop. If convicted, he could be jailed for one year or fined $1,000. (*St. Petersburg Times* 5/27/94 A3)

Computer Morphing Goes Political. First in TV commercials for shaving cream, now in TV commercials for politicians. Some of the credit for last week's election victory of a Kentucky candidate for U.S. Congress appears to go to a TV commercial that used computer morphing to identify the candidate's opponent with President Clinton, who is unpopular in the district. The commercial showed the opponent's face dissolve into the face of Mr. Clinton and then back again. (*Washington Week in Review, Public Broadcasting System*, 5/27/94)

Slashing Networking Costs. Silicon Valley upstart First Virtual Corp. says that by early 1995 it will deliver ATM (asynchronous transfer mode) networks for less than $500 a PC, considerably less than the $2,500 to $4,000 currently charged. First Virtual's networks will operate at a slower speed than some of its competitors, but will be compatible with existing networks and will use standard wiring. (*Wall Street Journal* 5/27/94 B5)

Not Your Father's Oldsmobile. *@Wired*, the name for *Wired Mgazine*'s new business venture, will be "part online service and part cyber-salon," according to New York Times writer John Markoff, but it won't look like Prodigy, America Online or Compuserve. Wired has its own view of the future. "The big sumo-wrestler corporations that are stumbling around trying to dictate the information superhighway have entirely missed the point," says Wired's publisher. "It's not about content, it's about connectivity." (*New York Times* 5/29/94 Sec.4 p.7)

Flaw Found in Clipper Chip. An AT&T computer scientist at Bell Labs says that a flaw in the Administration's Clipper Chip encryption technology (which is intended to let the Government eavesdrop, with a court order, on electronic digital communications). The flaw would allow expert computer users to defeat the government's ability to decode messages. The government's response: "Anyone interested in circumventing law-enforcement access would most likely choose simpler alternatives." (*New York Times* 6/2/94 A1)

Watch Out for Falling Prices. Expect to see 486-based PCs with CD-ROM drives selling for less than $1,000

in mass-merchandise outlets by year-end, and PDA prices dropping to $1,200 by 1995. (*Bottom Line Personal* 6/15/94 p.6)

Email at the White House. Both the Bush and Clinton administrations have tried to restrict public access to White House email, but later this year the National Security Agency will publish "White House EMail," a book-length collection of email messages. The book includes Iran-Contra affair communications to and from Oliver North, who used email because he thought it could be easily deleted. One message from him reads: "Oh lord. I lost the slip and broke one of the high heels. Forgive please. Will return the wig on Monday." (*So who needs the Clipper Chip?*) (*New York Magazine* 6/6/94 p.20)

Networked Gaming Now. Catapult Entertainment is marketing modems that will enable video-game players to compete with each other over existing telephone lines. (*Wall Street Journal* 6/6/94 B5)

Cybercop. A former New Jersey police officer now spends his time cruising for suspects in cyberspace and has been involved in dozens of criminal investigations, including a sting operation that nabbed a pedophile who lured young rape victims via a bulletin board service. (*Tampa Tribune* 6/8/94 BayLife 5)

Population Boom in Cyberspace. By the end of this year, nearly four million U.S. households will have signed on with one of the Big Three online services–American Online, CompuServe, or Prodigy. (*Investor's Business Daily* 6/9/94 A4)

Email Privacy. A survey of 301 companies by *MacWorld Magazine* showed that 78.4% of the firms do not spy on employee email or search their electronic files. Sen. Paul Simon (D-Ill.) has drafted legislation requiring that companies tell their workers that their email might be monitored. (*Tampa Tribune* 6/8/94 BayLife 5)

MS-DOS 6.0 and 6.2 Recalled. A federal judge is making Microsoft recall two versions of its MS-DOS operating system because they infringe on data compression. technology patents held by Stac Electronics. The judgment won't affect IBM or Compaq, but will mean the recall of a number of systems made by computer manufacturers who use MS-DOS 6.0 or 6.2 without having a license to use it from Stac. (*New York Times* 6/11/94 A1)

Commercial Mosaic Versions. Commercial versions of Mosaic will be launched this summer for large technology companies by Spyglass and for consumers by Quarterdeck. The Spyglass version of Mosaic will also be bundled into most major systems from Digital Equipment Corporation. Meanwhile, Marc Andreeson, the principal creator of NCSA Mosaic and now the co-principal of Mosaic Communications, Inc., says his company is working on a "next generation version, from the ground up, that will be commercially compelling." (*Internet Business Report* June 95, p.1)

An Update on Standards Relevant to USENIX Members

Nick Stoughton
USENIX Standards Report Editor
<nick@usenix.org>

This column used to be entitled "An Update on UNIX Related Standards Activities." However, UNIX is now a licensable trademark of X/Open. In order to be able to use it, a system must support the interfaces described in their SPEC 1170 document, otherwise known as XPG 4 release 2. I don't have all the 1170 interfaces, so I guess I shall have to rename the column "An Update on Standards Activities Relevant to USENIX Members." USENIX, of course, is not a trademark of X/Open. Thanks to Peter Salus for reminding me of the original occasion that this footnote was used.

Is SPEC 1170 the answer to a maiden's prayer? How will it affect the future of Open Systems and UNIX as we now it? 1170 has one major difference from its predecessors. It represents the union of all the current major implementations of UNIX and not the intersection. Earlier versions of the X/Open Portability Guide (XPG), while not standards as such, provided a list of interfaces that worked the same on all the X/Open branded systems. POSIX took that concept one step further, as well as making a real international standard. POSIX, though obviously derived from UNIX, has allowed non-UNIX people into the game. But to whom is X/Open accountable – the system vendors? Or the end users? Or their shareholders?

The strength of POSIX is that it is accountable to vendors, users, general interest, and academic groups. They all have power through the ballot process. The standard is based on consensus between these people. But the ownership of SPEC 1170 lies entirely with X/Open and its members. It is a tool that gives them considerable power. And you've all heard the old adage: "power corrupts, absolute power corrupts absolutely . . ."

Theoretically, because 1170 is a complete specification, missing only hardware specific interfaces, no one will ever need to use interfaces outside the specification. Both XPG and POSIX had "missing pieces"– interfaces that vendors could provide to make their system that little bit better. With SPEC 1170, there seems to be an attempt to say "no one will ever need any interfaces we haven't already thought of"– the end of systems programming in UNIX.

After several years, Jim Isaak is stepping down from the post of PASC (Portable Applications Standards Committee) chair. Jim's leadership has seen POSIX become one of the most important and respected standards in the world of Open Systems. I am sure you will all join me in wishing him well in his new role, defining standards for the National Information Infrastructure, or "Information Super-Highway."

His successor in PASC will be Lowell Johnson, of Unisys. Lowell is currently chair of the 2003 test methods groups, and has worked within POSIX since its inception. His vision is of a more efficient process, producing relevant standards faster.

The following reports are published in this column:

- POSIX.5: ADA Bindings
- The Distributed Service Working Groups
- SRASS: Services for Reliable, Available, and Serviceable Systems

Our Standards Report Editor, **Nick Stoughton**, welcomes dialogue between this column and you, the readers. Please send your comments to *<nick@usenix.org>*.

Report on POSIX.5: Ada Bindings

Robbie Robbins <robbie@lfs.loral.com> reports on the April 18-22, 1994 meeting in Lake Tahoe, Nevada:

The primary charter of the POSIX.5 group is to produce Ada language bindings to POSIX standards. The standard for Ada language bindings to 1003.1-1990 was published in 1992 as 1003.5-1992. The working group is now working on three projects:

- POSIX.5a – A few problem fixes to the 1003.5-1992 Standard

- POSIX.5b, comprised of the documents formerly named POSIX.20, the Ada binding to POSIX.1b, (the real-time extensions, otherwise known as POSIX.4) and Mutexes, Condition Variables and Thread I.D.s from POSIX.20a, the Ada binding to POSIX.1c (the real-time threads extensions of POSIX, otherwise known as POSIX.4a). One day, we will get used to these new numbers!

- P2003.5, the test assertions document for 1003.5-1992.

In addition, at the July meeting, the group will add a MOTIF/Ada project to produce an Ada Binding standard to 1295-1993, the Modular Toolkit Environment standard.

The POSIX.5 Interpretations Committee issued an interpretations document (1003.5-1992INT) in March containing responses to seven problems in 1003.5-1992 encountered by users of the standard. To give a flavor of the work, here is the list of titles and interpretation numbers:

1. Missing parameters from FLUSH_IO generic operations
2. Text on reading from a pipe
3. Text on writing to a pipe
5. Behavior of read when interrupted by a signal
7. Can IS_A_TERMINAL detect/report errors?
9. TEXT_IO files should not have EXECUTE rights by default

The committee also worked on a number of additional interpretations requests:

4. Error checking in POSIX_Configurable_File_Limits
8. Behavior of the Generic I/O Operations With Non_Blocking Option
10. Make Fork and Exec optional (rejected by the committee as a change)
11. File Pointers on Read/Write
12. Access time update on Generic_Read and Generic_Write
13. Blocking vs. nonblocking behavior on Read/Write

Some of these will require amendments to 1003.5-1992. The plan for POSIX.5a is to fix the known errors and rewrite Chapter 6 (Read/Write). The initial P1003.5a document should be ready for ballot after the July meeting. POSIX.5a, when issued, will be change pages, which will be merged into 1003.5-1992 with POSIX.5b after its approval.

POSIX.5b is the Ada binding to POSIX.1b (the real-time extensions of POSIX) and Mutexes, Condition Variables and Thread I.D.s from POSIX.1c (the real-time threads extensions of POSIX).

The first formal ballot on what was then named POSIX.20 was conducted on a "thin" binding version: that is, POSIX.20 did not repeat the underlying semantics of the POSIX Real-time Extensions draft, which is a C-language interface. The ballot showed that a "thick" binding version was clearly favored instead, not requiring references to the C version. Time since January has been spent importing the underlying semantics into the draft. This "thickening" process has in turn exposed problems in the bindings draft. In the time period before the April meeting, some of these problems were worked out and the document was edited for consistency.

Most of the April meeting consisted of group reviews and changes to the thick version, now renamed POSIX.5b, resolving the exposed issues in sections 1-12. The agreed changes to the document will be made between meetings and a final group review will be conducted during the July meeting. The object is to have a new draft ready by August 1 for ballot, scheduled for the month of September.

The POSIX.5 working group, together with the POSIX.4 working group, is still working to resolve the seven objections to POSIX.1c that the POSIX.5 working group submitted in July at a coordination ballot. Five of the objections are considered resolved, although the POSIX.5 group has not yet had the opportunity to review the text changes scheduled for POSIX.1c Draft 9. The remaining two objections are currently under negotiation between the chairmen of the two working groups. The first involves situations where the code of a signal handler needs to ensure that a mutex is locked. The other involves the change brought on by POSIX.1c from per-process signal masking to per-task signal masking.

After a period of funding uncertainty, DISA has provided funding to Jim Leathrum's group at Clemson University's Software Standards and Technology Laboratory to develop the test assertions for 1003.5-1992. The POSIX.5 working group appointed Kathy Liburdy as Vice Chair of POSIX.5 for Test Assertions. The Clemson group plans to produce a draft prior to September when the DISA funding expires. They are also on track to produce a description of their method as an appendix to P2003.

The IEEE Computer Society Portable Application Standards Committee authorized the formation of a group to standardize an Ada binding to 1295-1993, Modular Toolkit Environment (the IEEE standard for MOTIF.) This work was assigned to the POSIX.5 group. Dave Emery plans to sponsor a study group within POSIX.5, starting with an organizational meeting during the July POSIX.5 meeting. The POSIX.5 group concurred with this plan on the understanding that this work should not detract from any current POSIX.5 efforts. The group will need a Vice Chair, Secretary, and additional people dedicated to developing and balloting the proposed standard. The work should take two to four years.

A Tour of the Distributed Service Working Groups

David Cannon <D.Cannon@Exter.ac.uk> reports on the April 18-22, 1994 meeting in Lake Tahoe:

This Spring Tahoe was warm, dry and attractive, with snow on the surrounding peaks reflected in the waters of the lake. But even this scenery failed to attract the POSIX crowds, with attendance overall down to about 145–well below that of the previous meeting.

The Distributed Services groups contributed to the shortfall, with POSIX.8 and POSIX.12 not meeting at all, due to the unavailability of target documents or conflicts with close-of-ballot dates.

The groups' progress over the week is outlined below:

POSIX.8 (Transparent File Access). Progress is stalled on two counts: it's awaiting the production of the document merging both the POSIX.1 and POSIX.4 standards, (*Ed. Note:* the merged POSIX.1b document is now with the IEEE for reproduction and distribution) which the POSIX.8 work will further modify, and the recirculation ballot of its draft. This latter didn't happen on schedule as some of the `no' voters weren't contacted to confirm that their ballots had been resolved following the first round of ballot resolution.

The July meeting of POSIX.8 will work on the merge of their draft with the POSIX.1b document, which should by then be available, and a further ballot recirculation of the merged POSIX.1/.4/.8 (a.k.a. POSIX.1f) draft will take place following that.

POSIX.12 (Protocol Independent Interfaces). The group will be meeting separately in the week beginning 23 May. This schedule locks into the completion of the group's recirculation ballot, due on the 2 May. The group will resynchronize with PASC (the IEEE Portable Applications Standards Committee) in July. It was noted that X/Open are bringing their specifications in line with the sockets part of POSIX.12 via the X/Open fasttrack process. X/Open will track the changes introduced by the POSIX.12 ballot returns and introduce them to its work. The current POSIX.12 draft states a requirement for both sockets and XTI; this is echoed by the X/Open group.

POSIX.21 (Real Time Distributed Services). The first DS group working at Tahoe, which had up to twelve attendees through the week, and met for the full allotted time.

The group decided to pursue its proposed "Time Services" PAR as an addition to the POSIX.1. . . series of standards, rather than as an independent 13*xx* standard. The group is happy with the overall progress made, given that they are already working on Language Independent text, though this particular meeting had some uncharacteristically slow patches where it revisited old ground. The group plans to have a first draft available in July 1995, and currently intends first ballot for July 1996. The current intention is that all the drafts produced by the group will take the "thick" form.

POSIX.22 (Security Framework Guide). The group met together with the available POSIX.6 people, the POSIX.6 draft being in ballot, (closing on 18 May) and had eight people in over the week. The current draft went out in the March POSIX mailings. The first day was spent reviewing the document from the viewpoint of the anticipated audiences. These fall into two groups; security aware and security unaware(!). This revelation determined the group to restructure the guide, and an executive overview to the draft was crafted during the week.

Steps towards ballot group formation will be made following the Tahoe meeting, with ballot projected to follow the July meeting if the schedule permits.

1238 (FTAM and OSI Services API). The group had three people attending. Their 1351 and 1353 (OSI API) drafts have completed ballot recirculation with 94% approval. By the Monday evening the group had resolved most of the remaining objections, one of which involved substantive changes to 1351–thus requiring a third recirculation ballot, scheduled for 5 July. The hope is to reach the September meeting of the IEEE Standards Board for approval.

1238.1 (FTAM) drafts were available at the meeting. The ballot window is set for July. Closing date for the formation of the ballot group is scheduled for 3 May, but the group is very keen to gain more members of the ballot group and its close will be delayed for as long as possible. The first ballot is targeted for 30 days from 6 June.

The supporters of the ROSE API proposal, notionally targeted for the 1238 group, have not followed up with a Project Authorization Request (PAR) or any other expression of interest. This may die unless some champion (X/Open perhaps?) comes forward to introduce their ROSE API to PASC, possibly through the IEEE fasttrack process.

Report on SRASS: Services for Reliable, Available, and Serviceable Systems

Arun Chandra <achandra@vnet.ibm.com> reports on the April 18-22, 1994 meeting in Lake Tahoe, Nevada:

Are you interested in Fault Tolerance, High Availability, Reliability, Serviceability, Maintainability? If so consider joining the "Fault Management Study Group" at the next POSIX meeting at Nashua, NH in July. The group approved a name change to "Services for Reliable, Available, and Serviceable Systems Study Group." If you see any reference to either of the above two names, its us.'

This group first met in October '93, following BOF sessions at the two previous meetings. The status of the group is a "Study Group" preparing a "Project Authorization Request" (PAR). The PAR will go up for review at the July meeting. If approved we will become an official POSIX working group. Healthy participation at the July meeting would indicate that Fault Management is something organizations are interested in seeing standardized. This is one of the basic criteria for PAR approval, especially in these hard times.

A number of existing documents are being studied as base documents. To obtain a list of these documents or the documents themselves please contact the chair of the group – Helmut Roth <*hroth@relay.nswc.navy.mil*>. The detailed minutes of the October '93, January '94, or April '94 meetings can also be obtained from Helmut.

The primary task of the working group at the April meeting was to get a PAR approved. A draft was submitted to the PAR Management Subcommittee (PMC), which was approved for recommendation for sponsorship. However, to get this approval, the study group felt that the scope of the standard had been overly reduced. As a result at the Sponsor Executive Committee (SEC) meeting the PAR was not sponsored and will be revised and resubmitted at the July meeting.

The group felt that the PAR's scope must include two major areas:

- Fault-tolerant issues, and
- Serviceability issues.

There was a debate as to whether the there was too much initial concentration on the Serviceability issues. The Fault-Tolerance community representatives at the group, who are actually the majority, want to strike a good balance between the two areas. IBM's AIX documentation has been identified as the base document for serviceability issues, but no base document has been identified as yet for fault tolerant issues. All the above issues will be reflected in the revised PAR.

This study group spent an intensive week in the PAR approval and revision process. The group is optimistic that the PAR will be approved in July.

If you are interested in more information on the group, why not contact the group Chair, Helmut Roth, or me, the group Vice-Chair.

The Bookworm

by Peter H. Salus
<peter@uunet.uu.net>

After listening to Steve Bellovin's presentation at USENIX in Boston, I went back and reread part of the *Firewalls* book (reviewed by me and by Rik Farrow in the previous *;login:*). Not only is it a good book, but Rik's wrong in his remark about the `relay` source.

Distributed systems

"Distributed Systems" is a marketeer's buzzword; but it's also a genuine phenomenon in computing. Coulouris and his colleagues at Queen Mary (and Westfield) College have turned out a textbook that's worth reading by those not in (or teaching) the class. While I enjoyed the introductory characterizations of the concept, I thought that the chapters on Networking and Internetworking (3), Name Services (9), security (16), distributed shared memory (17), and the 60 pages of case studies (Mach, Chorus, Amoeba, Clouds, etc., and a comparison of Mach, Amoeba and Chorus) are really good.

This is an outstanding book: I thought I'd read a chapter and put it aside. Believe me, gentle reader, I read it all.

cat -v, again?

At the Toronto USENIX in June 1983, Rob Pike railed at the bloat of the UNIX Operating System, citing BSD's cat -v as an example. Perhaps this same bloat is now afflicting books on the OS.

Bach's 1986 book on SVR2 ran to under 500 pages, as did Leffler et al. on 4.3BSD three years later. Goodheart and Cox on SVR4 is nearly 700 pages. Many of them are, indeed, good pages. But there sure are a lot of them.

The Magic Garden Explained is a book you can rely upon. It is a solid book. Though there are slips in the history section, it is better than most by far. I found Leffler, et al., a difficult book to read, but it was definitely well worth reading. Goodheart and Cox is not gripping, but I note that it's not the first, either. It is also printed in a decidedly ugly fashion: perhaps Prentice Hall needs a designer.

The *Garden* is arranged differently from either of the other volumes on UNIX internals, but it is a useful organization. It is also far more overtly a textbook than the others, though both of them contained exercises, too.

Don't get me wrong: I admit to not being a fan of SVR4. But, this is a worthwhile book. If I were teaching an SVR4 course, I'd use it as a text. But there are parts of it that seem strange to me. Why for example print the Appendix (F) full of crash commands? The multi-page table is only typographically different from the online man page.

Until Leffler, et al., is upgraded, though, this is the best work on current UNIX internals around. Perhaps I shouldn't have just read it; it might have worked better as a reference or a guide that I dipped into.

Books reviewed in this column:

George Coulouris, Jean Dollimore and Tim Kindberg, **Distributed Systems: Concepts and Design**. 2nd ed. Addison-Wesley, 1994. 644pp. ISBN 0-201-62433-8. $49.50.

Berny Goodheart and James Cox, **The Magic Garden Explained**. Prentice Hall, 1994. 664pp. ISBN 0-13-098138-9.

Andy Gaffin, **Everybody's Guide to the Internet**. MIT Press, 1994. 211pp. ISBN 0-262-671057. $14.95.

Harold Salzman and Stephen R. Rosenthal, **Software by Design**. Oxford University Press, 1994. 348pp. ISBN 0-19-508340-7. $39.95.

Douglas E. Comer and David L. Stevens, **Internetworking with TCP/IP: Vol. 2: Design, Implementation, Internals.** 2nd ed. Prentice Hall, 1994. 612pp. ISBN 0-13134677-6. $50.

Ed Krol, **The Whole Internet User's Guide and Catalog**. 2nd ed. O'Reilly & Associates, 1994. 544pp. ISBN 1-56592-063-5. $24.95.

Lowell Jay Arthur and Ted Burns, **UNIX Shell Programming**. 3rd ed. John Wiley & Sons, 1994. 462pp. ISBN 0471599417. $29.95.

The X Resource Issue 10. O'Reilly & Associates, 1994. 198pp. ISSN 1058-5591. $22.50.

Yet More Internet

Gaffin's *Everybody's Guide to the Internet* is what you used to be able to ftp from the EFF as *The Big Dummies' Guide...* I guess MIT Press didn't want to call it that. It's a B+ job, at under 200 pages of text, and a reasonable price. There's a good glossary, too. But: once again there's a list of providers. It is far from complete. I was asked by someone I showed the book to whether the EFF had a ``good guys'' and a ``bad guys'' list. I answered no. But that's the kind of impression that's given (Netcom, for instance, is nowhere listed).

I was also mildly puzzled by the way addressing was handled in the chapter on mail. Where Delphi, PeaceNet and others are concerned, the reader is told to use (e.g.) "username@delphi.com." For me, would this be peter@. . . , peter_h_salus@. . . peter_salus@. . . , salus@. . . peter.salus@. . . , or what?

I counted 120 books with Internet in the title on a recent visit to a large bookstore. This did not include such items as Comer/Stevens or Rose/Lynch. I don't know that we need many more. This one does have a nice Foreword by Mitch Kapor and mercifully does not come with ads for America Online or Delphi or . . . .

Design Features

I was relatively excited by *Software by Design* when I started reading it. I thought that perhaps I would gain a good deal of knowledge about "Software in the Workplace" and the "Structure of Software Design." Not! This is an academic book by two academics (Boston University and University of Louisville) and published by an academic press. It is full of sentences like: "User-driven design approaches provide greater assurances that software designs will meet the needs of users and the requirements of the work process" [p. 73]. Yep. Pp. 275-305 concern "Keeping the Customer Satisfied: Field Service and the Art of Automation." Perhaps everyone could send in a card naming the company with the service organization most in need of this.

Re-issues

There is a new second edition of Volume 2 of Comer and Stevens. The 1991 edition was 80 pages shorter. The new edition has OSPF and IGMP. I presume that there will be new editions of both versions of volume 3, too. Collect 'em all!

ORA has brought out a second edition of *The Whole Internet*. Ed Krol should be very proud.

There is a third edition Arthur and Burns' *UNIX Shell Programming*. This is a good book; but it is limited: it has good coverage of the Bourne, C and Korn shells. Neither bash nor tcsh is mentioned at all. Nor is POSIX compliance.

Issue Ten of *The X Resource* is out. I found Thomas Phelps' "TkMan" piece – about a *man* page browser based on Tk – most valuable.

Ethical Conflicts in Information and Computer Science, Technology, and Business

by Donn Parker, Susan Swope, and Bruce N. Baker. QED Information Sciences, Inc., 1994, 245 pp. ISBN 0-89435-313-6.

Reviewed by Rob Kolstad
<kolstad@bsdi.com>

I dealt with this book in pre-press many years ago. Now it is available in softcover. One must realize that the world has changed since the participants in the 1987 workshop met – people are far more computer literate. The potential for abuse, only imagined in 1987, can now be implemented with ease.

In 1970 a group of professionals and ethicists discussed computer ethics and wrote a report on it. In 1977 an updated meeting was held. Using the 1977 paradigm, this book examines the discussions and ideas from a 1987 meeting. Each chapter focuses on a theme in the world of business (e.g., Professional Standards and Accountability, Property Ownership, Employee Relations). The "panel of experts" is presented with a scenario. Here's one I've summarized from the accountability chapter:

The owner of a computer dating service finds that he is being sued by customers who say they contracted AIDS by meeting people through the dating service. They want the service to be terminated. He says, "No, there's always a risk; it's not my responsibility to warn you or determine health risk factors."

The panel is asked to choose among unethical, not unethical, or no ethics issue present. Subgroups analyze the problem, impressions are tallied, and a nice table is drawn up. The analyses are particularly enjoyable, because they are written clearly and always begin with "first principles." I like that. The scenarios are interesting and realistic. In fact, I've been in an incident quite similar to one of those presented. Regrettably, I came up on the unethical side of the issue and ultimately the employee involved was incensed. I now think more about ethics and feel better about the environments in which I work and in which I deal with others.

Like Dale Carnegie's tome *How to Win Friends and Influence People*, I expect this is a book that bears re-scanning every few years. I don't know how each scenario will stand the test of time. I'd be fascinated to purchase a follow-up study done in the near future, for computers have proliferated widely and people's expectations and perceptions of them have come to vary widely as well.

If you're into ethics, this is the book for you.

Internetworking with TCP/IP, Volume I: (Second Edition) Principles, Protocols, and Architecture

By Douglas E. Comer, Prentice Hall,1991, 547pp.
ISBN: 0-13-468505-9

Internetworking with TCP/IP, Volume II: Design, Implementation, and Internals

By Douglas E. Comer and David L. Stevens Prentice Hall, 1991, 532pp. ISBN: 0-13-472242-6

Internetworking with TCP/IP, Volume III: Client-Server Programming and Applications

By Douglas E. Comer and David L. Stevens, Prentice Hall, 1993, 498pp. ISBN: 0-13-474222-2

Reviewed by George V. Neville-Neil
<gnn@netcom.com>

In 1988 the first edition of *Internetworking with TCP/IP* (Volume I) appeared in my local technical bookstore. I had been looking for a single, concise, reference on the Internet for a year or so but had so far been unable to find one. I'd read some of the RFCs but they were not enough to explain the Internet to a rank novice. Prior to this I had used the Comer books on Xinu in my Operating Systems courses, and had enjoyed them. I went back a couple of times before buying the book: it probably cost about $40.00, which was not a small sum to a college student. Finally, I bought the book and took it home to read.

Over the next few weeks I carried that book with me everywhere, and read it whenever I got the chance. I was completely engrossed by it. The writing was smooth and kept my attention, even though the subject matter would probably be considered far from interesting by most people.

I now own two sets of the books, one for my home and one for my office. The second two volumes are just as enjoyable to read as the first, written in the same clear and concise style.

The first volume deals with the principles behind the TCP/IP protocols. It works its way from the lowest levels (discussing Local Area Networking, Metropolitan Area Networking, and Wide Area Networking in chapter 1) up through the base level Internet Protocol. It then goes on to the higher levels of UDP and TCP protocols. Several chapters are dedicated to routing protocols, followed by a discussion of the socket interface. The final chapters discuss applications, such as TELNET, rlogin, FTP, and SMTP (Mail). Each chapter contains diagrams, including pictorial representations of packet structures.

A particularly helpful feature is that important concepts are presented in a few sentences in italics to highlight them. Thus, page 62 in the chapter on Internet Addresses: "Each host on a TCP/IP internet is assigned a unique 32-bit internet address that is used in all communication with that host." Setting these comments apart not only makes the book easily accessible for skimming, but makes sure that the intended message is communicated to the reader.

Each chapter ends with a summary of all of the concepts presented in that chapter, a section on Further Study, and a set of exercises, ranging from simple questions whose answers are readily available in the chapter, to suggestions for small projects.

The book has several appendices, including a Guide to RFCs, a Glossary of Internetworking Terms and Abbreviations, a discussion of Standardization and Examples of Internet Information Archives. The bibliography is excellent, and the 13 page index makes finding answers to specific questions quite easy.

As an introduction to the TCP/IP protocol suite, and its theoretical underpinnings, this is an excellent book. The exercises render it serviceable for teaching a college level course on networking, with practical examples and applications. It is also a good reference book to keep around for anyone who is working with TCP/IP.

Volume II (Design, Implementation and Internals) is a study of one implementation of the IP protocol suite. The operating system used is Xinu, a UNIX-like operating system used for teaching, and that has also been used as an OS for embedded systems.

The text works its way up from "The Structure of TCP/IP Software in An Operating System," in chapter 2, through "Address Discovery," "Routing," and finally through several chapters on TCP and SNMP. The last chapter is a retrospective of the implementation that was presented in the book.

As a teaching tool the second volume is probably even better than the first. It is aimed at an audience that is trying to learn the TCP/IP protocols, and understand their inner workings. There is a great deal of example source code, drawn from the working system, that is used to explain the concepts discussed. This volume also contains a complete bibliography, and an extensive index.

The third volume (Client-Server Programming and Applications) presents a complete set of applications that use a client/paradigm of communication. Beginning with simple client services, such as DAYTIME and ECHO, and moving through several levels of complexity (including concurrent, connectionless, and connection oriented services) we come to a complete description of XDR (eXternal Data Representation) and RPC (Remote Procedure Call). Following these application protocols are several chapters that present a discussion of NFS and a TELNET client.

Example source code is used extensively throughout the book, and the first appendix is a complete description of all of the system calls relating to sockets. This book is an excellent reference for anyone using Sun RPC (aka ONC RPC) as it has one of the best explanations of RPC and XDR and their compilers (rpcgen) that I have read.

For people building distributed applications using the TCP/IP protocols this book is a must. It contains all of the information you need to get started in one place, and remains a good reference for experienced network programmers as well. Those of you with anonymous ftp access on the net can also get the source code used in this book. It's available on: *arthur.cs.purdue.edu* in */pub/dls*.

It should be obvious by now that I like all three of these books. The first and the third volumes are excellent references, as well as teaching tools, while I find the second to be more firmly in the teaching arena. I purchased my office copies as a set, and I keep them handy. I think they're an excellent asset for those who have to deal with the TCP/IP protocol suite.

Connecting to the Internet

By Susan Estrada. An O'Reilly Buyer's Guide; O'Reilly & Associates, 1993. 170 pp. ISBN 1-56592-061-9. $15.95.

Reviewed by Adam S. Moskowitz
<adamm@world.std.com>

This book sets out to help folks get connected to the Internet, and it does just that. It's written for people who know a little about networks, but don't really understand the myriad of ways the Internet can be used or what's the best type of connection for their purpose. Estrada starts out with clear, concise explanations of the basics, then goes on to compare and contrast the different types of connections.

The first three chapters cover how the Internet works, network performance, and how the Internet can (and will) be used. In this last chapter, Estrada not only explains each of the common uses of the net, she also illustrates the load each use places on the connection. The chapter ends with a concise list of questions to be answered before approaching a network provider. Having the answers to these questions will make selecting a provider much simpler, and will force readers to think about what they require from the net.

The fourth chapter provides insight on choosing a network provider. Here, Estrada covers not only technical issues such as services and speeds offered, but references, user support, and rapport. Again, the chapter ends with a list of questions to be considered when selecting a provider.

The last two chapters cover dial-up and dedicated connections, with dial-up further divided into online accounts, SLIP/PPP, and X.25/ISDN. While the explanations and discussions are good, the most useful part of these two chapters is the cost comparison charts. Here, Estrada gives the reader a place to compare hard numbers, although the reader has to obtain the numbers. With costs such as set-up fees, fixed and variable monthly and hourly charges, and toll costs filled in, the real cost of a connection can be seen, and connections from different provider compared. As before, these chapters each end with a list of questions to consider before deciding on the type of connection and the provider.

The appendices take up over half the book. Appendix A is the PDIAL list: Peter Kaminski's Public Dialup Internet Access List. Even though the list is likely to be out of date, it's better than nothing for the folks that don't already have ftp access. Appendix B is a list of Internet providers, both in the States and abroad. Again, the list is out of date, but it's a good starting point. One minor glitch – if you happen to be looking for a connection in Sweden, SWIPnet, the SWedish IP Network, is in the alphabetical list of U.S. providers. O'Reilly will be fixing this in a future edition.

While I'm talking about glitches – the first edition (August, 1993) has some problems with the use of italics. In some places, the italicized word overlaps the preceeding word. Again, O'Reilly knows about this, and has promised to fix it, though there's no date for the new edition yet.

Estrada's book may seem overly simple to many of us; for the folks who haven't been on the net since "way back when," it's what's needed. The charts and lists of questions distill the information in the chapters down to the essentials when choosing the type of service and the provider. The glossary has good explanations of the words and TLAs many

of us take for granted, and the bibliography has good pointers for new folks. At $15.95, it's money well spent.

;login: 50 and 100 Years Ago

by Barry Shein
<bzs@world.std.com>

August 1944

A question I am often asked is: Is data necessary? Well, to be truthful I have only been asked this once. And even then only after I engaged in some frantic prodding. Nevertheless, after having been asked even once, the question deserves an answer. As a Dr. Twelvepunch wrote many years ago: "Data, data, data." He then promptly left his wife, children and a rather odd dog named Hieronymous for a new career selling nutburgers somewhere in the vicinity of Huntington Beach, California. This was the first recorded case of *data anxiety.* One cure for data anxiety is simply never to think about data. However, many of us in this profession find that impossible to do, or at least impossible to do for very long. Once one has finished sorting the crayons first according to the color spectrum and then again alphabetically, checking several times that the gas jets are off on the stove and the pilots properly lighted and verifying that the capital of South Dakota is indeed Pierre data inevitably returns to dominate one's thoughts.

A man living in Tribeca, NY (who shall remain nameless) is reported to have put data out of his mind for several hours only to find that the reward was a particularly disturbing tax audit and several long-forgotten relatives suddenly calling him more frequently. Whether or not this can be asserted with confidence to be a case of cause and effect remains a mystery but few of us once warned would dare to tempt fate again in this manner.

Is Data Necessary?
Lt. Col. H. R. L. LeBoutellier, C.I.E.
Proceedings of the Third USENIX Symposium on Peculiar Musings

August 1894

Taking a page from modern physics we propose that research be carried out to assess the viability of sending messages through the now well-understood *ether.* This ether which binds together all of time and space will become a medium of the future replacing the unwieldy wires which at present transmit voice and telegrams.

Essentials of Wire Administration
Emma Nemeth
Tutorial Notes, USENIX LISA Workshop

We have developed a method for imprinting the tops of gentlemen's bathing suits with artwork at long distances via wire transmissions. Samples of this work with our demon emblem should be available for sale at the upcoming USENIX conference.

Letter in *;login:*
Kendall McKusick

!%@:: A Directory of Electronic Mail Addressing & Networks (4th Edition)

By Donnalyn Frey & Rick Adams, O'Reilly & Associates, 1994, 662 pp. ISBN: 1-56592-046-5, $9.95 (US)

O'Reilly just released the fourth edition of this basic guide to electronic mail. One significant change is that price has been lowered from $24.95 to $9.95 (yep, $9.95).

It includes a general introduction to the concept of electronic mail (e-mail) addressing, followed by a detailed reference section that describes how to address e-mail messages so that they can be sent from one network to another. It also includes a directory of hundreds of second- and third-level domains so users can address individual sites within a network. In addition to updating information about networks listed in previous editions, this new edition provides information about new networks in the Commonwealth of Independent States, Eastern Europe, Africa, and South America.

For more information, contact Brian Erwin of O'Reilly and Associates at *<brian@ora.com>*.

LISA VIII

8th USENIX Systems Administration Conference
San Diego, California
September 19-23, 1994

Tutorial Program
Monday - Tuesday, September 19 - 20, 1994

USENIX Association's well-respected tutorial program offers you introductory as well as advanced, intensive yet practical tutorials. Courses are presented by skilled teachers who are hands-on experts in their topic areas. The USENIX Systems Administration tutorial program has been developed to meet the needs of an audience of novice through experienced computer professionals.

Attend the tutorials at San Diego and benefit from this opportunity for in-depth exploration and skill development in essential UNIX system administration topics. Combining the two-day tutorial program with the three days of technical sessions means you have an opportunity to learn from experts at a convenient time and at a reasonable cost.

The tutorial program at San Diego is divided into five tracks – two full-day and six half-day tutorials on each day. Attendees may select any non-overlapping set of classes. Continuing in the LISA tradition, students will receive class notes for all the tutorials on the day(s) that they select. However, to ensure adequate seating and to reduce crowding, we are requiring that students pre-register for specific classes. Please note that some prior knowledge is required for the advanced tutorials. The USENIX tutorial program continues to enjoy a very high demand for its offerings. Several tutorials sell out before registration closes. Attendance is limited, and pre-registration is strongly recommended. On-site registration is possible ONLY if space permits.

Continuing Education Units

The USENIX Association is now a provider of Continuing Education Units (CEUs). Established by the International Association for Continuing Education and Training, the CEU is a nationally recognized standard unit of measure for continuing education and training, used by thousands of organizations across the United States. USENIX offers CE credit for its tutorials for a small additional administrative fee. Completion of any full day (or two half day) USENIX tutorial(s) qualifies you for 0.6 CEUs. We provide a certificate for each attendee taking a tutorial for CE credit and maintain transcripts for all CE students. You may request CE credit by checking the appropriate box on the registration form.

Monday Tutorials

Topics in System Administration
Trent Hein, XOR, and Evi Nemeth, University of Colorado, Boulder

Advanced Solaris System Administration
Peter Galvin, Brown University, and Dinah McNutt, Consultant

Internal Security
Rob Kolstad, BSDI, Inc.

Managing Time and Talent
Tina Darmohray, Lawrence Livermore National Laboratory, and Rob Kolstad, BSDI, Inc.

DNS Management
Tina Darmohray, Lawrence Livermore National Laboratory

Writing Safe SUID Code
Tom Christiansen, Consultant

Managing a DCE Environment
Jonathan Chinitz, IntelliSoft Corp.

Legal Issues for System Administrators
Dan Appelman, Heller, Ehrman, White & McAuliffe

Tuesday Tutorials

More Topics in System Administration
Trent Hein, XOR, and Evi Nemeth, University of Colorado, Boulder

Joining the Internet Safely using UNIX and Firewalls
Tina Darmohray, Lawrence Livermore National Laboratory

Advanced Sendmail Configuration
Rob Kolstad, BSDI, Inc.

Ethics & Policies
Rob Kolstad, BSDI, Inc.

The Perl Toolbox–Perl by Example
Tom Christiansen, Consultant

NIS+
Marc Staveley, Consultant

The Canons of Computer Services Management
William E. Howell, Glaxo, Inc.

Performance Management and Help Desk Management
Barb Dijker, Consultant

Technical Program

"Automation: Managing the Computer of the 90s"

Wednesday – Friday, September 21-23, 1994

Wednesday, September 21

9:00-10:30
Keynote
Jack Stanley, Houston Chronicle

11:00-12:30 **Track I**
System Configuration
Session Chair: *Mark Verber, Xerox PARC*

Central System Administration in a heterogeneous UNIX Environment: GeNUAdmin, *Dr. Magnus Harlander, GeNUA mbH*

Config: A Mechanism for Installing and Tracking System Configurations, *John P. Rouillard, Richard B. Martin, University of Massachusetts – Boston*

Towards a High-Level Machine Configuration System, *Paul Anderson, University of Edinburgh*

11:00-12:30 **Track II**
Invited Talk: Purchasing a Workstation Installation, *Peter Galvin, Brown University*

2:00-3:30 **Track I**
Automation
Session Chair: *Hal Stern, Sun Microsystems*

OMNICONF – Make OS Upgrade and Disk Crash Recovery Easier, *Imazu Hideyo, Matsushita Electric*

Automated Upgrades in a Lab Environment, *Paul Riddle, University of Maryland*

Tenwen: The Reengineering of a Computing Environment, *Remy Evard, Northeastern University*

2:00-3:30 **Track II**
Invited Talk: Plenty of Confusion (PC) on the Network, *Craig Hunt, National Institute of Standards and Technology*

4:00-5:00 **Track I**
The Toolbox
Session Chair: *Pat Parseghian, AT&T Bell Laboratories*

Kernel Mucking in Top, *William LeFebvre, Argonne National Laboratory*

Handling Passwords with Security and Reliability in Background Processes, *Don Libes, National Institute of Standards and Technology*

4:00-5:30 **Track II**
DCE's Impact on System Administrators
Rich Salz, Open Software Foundation

Thursday, September 22

9:00-10:30
Keynote: Breaking into Banks
Dan Geer, OpenVision Technologies

11:00-12:30 **Track I**
Software Configuration
Session Chair: *Paul Evans, Synopsis, Inc.*

Soft: A Software Environment Abstraction Mechanism, *Remy Evard and Robert Leslie, Northeastern University*

Beam: A Tool for Flexible Software Update, *Thomas Eirich, University of Erlangen-Nürnberg*

Depot-Lite: A Mechanism for Managing Software, *John P. Rouillard, Richard B. Martin, University of Massachusetts – Boston*

11:00-12:30 **Track II**
Invited Talk: Network Security Fun
Bill Cheswick, *AT&T Bell Laboratories*

2:00-3:30 **Track I**
Works-in-Progress

2:00-3:30 **Track II**
Invited Talk: Performance Monitoring and Tuning, *Marc Staveley, Consultant*

4:00-5:30 **Track I**
Automation, the Sequel
Session Chair: *Neil Todd, Swiss Bank Corporation*

SENDS, a Tool for Managing Domain Naming and Electronic Mail in a Large Organization, *Jerry Scharf, Sony Corporation and Paul Vixie, Vixie Enterprises*

Getting More Work Out Of Work Tracking Systems
Elizabeth D. Zwicky, Silicon Graphics

Managing the Ever-Growing To Do List, *Remy Evard, Northeastern University*

4:00-5:30 **Track II**
Invited Talk: A Practical Introduction to SNMPv1, *Phil Draughon, Northwestern University*

Friday, September 23

9:00-10:30 **Track I**
;login:
Session Chair: *Trent Hein, XOR*

Speeding Up UNIX Login by Cacheing the Initial Environment, *Carl Hauser, Xerox Palo Alto Research Center*

THE BNR/NT Standard Login (A Login Configuration Manager), *Christopher Rath, Bell-Northern Research*

Exporting Home Directories on Demand to PCs, *David Clear, Alan Ibbetson, University of Kent at Canterbury, Peter Collinson, Hillside Systems*

9:00-10:30 **Track II**
Invited Talk: Building a Successful WWW Server, *Amy Kreiling, University of North Carolina*

11:00-12:30 **Track I**
Peek a Boo–I Can See You
Session Chair: *William LeFebvre, Argonne National Laboratory*

Monitoring Usage of Workstations with a Relational Database, *Jon Finke, Rensselaer Polytechnic Institute*

Adventures in the Evolution of a High-Bandwidth Network for Central Servers, *Karl L. Swartz, Les Cottrell, Marty Dart, Karl Swartz Stanford Linear Accelerator Center*

Pong: A Flexible Network Services Monitoring System, *Helen E. Harrison, Mike C. Mitchell, Mike E. Shaddock SAS Institute, Inc.*

11:00-12:30 **Track II**
Invited Talk: A Sysadmin's Guide to SCSI: A Nonengineering Perspective or What the Heck is a Terminator? *Michael Pearlman, Rice University*

2:00-3:30 **Track I**
The Automation Revolution
Session Chair: *Tom Christiansen, Consultant*

Automating Printing Configuration, *John Finke, Rensselaer Polytechnic Institute*

Highly Automated Low Personnel System Administration in a Wall Street Environment, *Harry Von Hamhorse Kaplan, Sanwa Financial Products*

The Group Administration Shell and the GASH Network Computing Environment, *Jonathan Abbey, University of Texas – Austin*

2:00-3:30 **Track II**
Invited Talk: Getting on the Mbone, *Steve Casner, University of Southern California*

USENIX CONFERENCE ON OBJECT-ORIENTED TECHNOLOGIES (COOTS)

JUNE 26–29, 1995
MONTEREY, CALIFORNIA

ANNOUNCEMENT & PRELIMINARY CALL FOR PAPERS

The COOTS conference is designed to be a showplace for advanced development work in object-oriented technologies. The conference will emphasize research and experience derived from efforts to use object-oriented techniques to builds complex systems that meet real world needs.

The COOTS conference will begin with two days of tutorials. The tutorial program will offer a selection of tutorials from among several tracks. We expect tutorial topics to include:

- distributed object systems (CORBA, etc.)
- object-oriented network programming
- alternative object-oriented languages
- advanced techniques in memory management
- efficient and effective class design

Two days of technical sessions will follow the tutorials. Proceedings of the conference will be published by USENIX and will be provided free to technical session attendees; additional copies will be available for purchase from USENIX.

Like the USENIX C++ Conferences and Advanced Topics Workshops from which it is derived, COOTS will emphasize the advanced engineering aspects of object technology. While papers covering work in C++ are encouraged, the conference is broader in scope than its ancestors and invites submissions describing results and work in other object-oriented or object-based languages.

DATES FOR REFEREED PAPER SUBMISSIONS

- Submissions due: March 6, 1995
- Notification to authors: April 3, 1995
- Camera-ready final papers due: May 15, 1995

PRELIMINARY PROGRAM COMMITTEE

- Program Chair: Vincent F. Russo, *Purdue University*
- Tutorial Program Chair: Doug Lea, *SUNY Oswego*

Mark Linton, *Silicon Graphics*
Chris Pettus, *Taligent*
Jim Waldo, *SUN Microsystems*
Murthy Devarokonda, *IBM Watson Research Labs*
Addtional members to be announced.

CONFERENCE TOPICS

We seek papers describing original work concerning the design, implementation, and use of object-oriented technologies. Questions regarding a topic's relevance may be addressed to the program chair via electronic mail to *russo@cs.purdue.edu*.

Potential topics include:

- work on object-oriented programming languages (C++, Modula-3, Eiffel, etc.)
- implementations of commercial object infrastructures (CORBA, NextStep, OLE-II, SOM/DSOM, etc.)
- interface description languages
- distributed object systems
- unique applications of and experiences with object-oriented technologies

REFEREED PAPER SUBMISSIONS

Submissions must be received by March 6, 1995. Full papers should be 10 to 15 pages. Instead of a full paper, authors may submit an extended abstract which discusses key ideas. Extended abstracts should be 5–7 pages long (about 2500–3500 words), not

counting references and figures. The body of the extended abstract should be complete paragraphs. The object of an extended abstract is to convince the reviewers that a good paper and presentation will result. While, by acceptance of extended abstracts, we intend to stimulate industrial participation, submission of extended abstracts by academics is in no way discouraged.

All submissions will be judged on originality, relevance, and correctness. Each accepted submission will be assigned a member of the program committee to act as its shepherd through the preparation of the final paper. The assigned member will act as a conduit for feedback from the committee to the authors. Camera-ready final papers are due May 15, 1995.

Please accompany each submission with a cover letter stating the paper title and authors along with the name of the person who will act as the contact to the program committee. Please include a surface mail address, daytime and evening phone number, and, if available, an email address and fax number for the contact person.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1-510-528-8649, or email to *cootsauthors@usenix.org* or to the program committee chair.

The COOTS conference, like most conferences and journals, requires that papers not be submitted simultaneously to another conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non-disclosure agreement" forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

Where To Submit

Please send one copy of a full paper or an extended abstract to the program committee via one of the following methods. All submissions will be acknowledged.

- Preferred Method: email (Postscript or ASCII) to *cootspapers@usenix.org*
- Alternate Method: postal delivery to
  USENIX COOTS Conference
  c/o Dr. Vincent F. Russo
  Department of Computer Sciences
  Purdue University
  West Lafayette, IN 47907 USA
  Telephone: +1-317-494-6008

Dates For Refereed Paper Submissions

- Submissions due: March 6, 1995
- Notification to authors: April 3, 1995
- Camera-ready final papers due: May 15, 1995

For Registration Information

Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be available beginning in April 1995. If you wish to receive the registration materials, please contact USENIX at:

- USENIX Conference Office
  22672 Lambert Street
  Suite 613
  Lake Forest, CA 92630
  USA
- Telephone: +1-714-588-8649
- Fax: +1-714-588-9706
- Internet: *conference@usenix.org*

2nd USENIX Symposium on Mobile and Location-Independent Computing

APRIL 10–11, 1995
ANN ARBOR, MICHIGAN

ANNOUNCEMENT & PRELIMINARY CALL FOR PAPERS

The Second USENIX Symposium on Mobile and Location-Independent Computing will provide a major opportunity for researchers and practitioners in this rapidly growing field to exchange ideas and present results of their work.

The First Mobile Computing Symposium, held in Boston in August 1993, generated a great deal of interest from the UNIX and mobile computing communities. Since that time, mobile computing has become an even hotter topic, with the size, cost, and power requirements of the equipment going down. The FCC has announced a plan to auction radio spectrum for use of mobile devices, and the Internet Engineering Task Force (IETF) is in the process of standardizing protocols for mobile TCP/IP, including roaming capabilities. Mobile computers are the fastest growing segment of the PC market, airlines are scrambling to provide network connectivity on board, and terminal rooms at computer conferences routinely provide network taps for users who bring their own computers.

The 1995 symposium is a single-track symposium offering two days of refereed paper presentations. The symposium will also include panels, Work-in-Progress reports, Birds-of-a-Feather sessions, and a Keynote speaker. Formally reviewed papers, presented during the symposium, will be published in the symposium proceedings. Proceedings will be distributed free to attendees and later will be available for purchase from the USENIX Association.

Dates For Refereed Paper Submissions

- Extended abstracts due: January 2, 1995
- Notification to authors: January 23, 1995
- Camera-ready final papers due: March 6, 1995

Program Committee

- Program Chair: Jim Rees, University of Michigan
  Jim.Rees@umich.edu

Dan Duchamp, Columbia University
djd@cs.columbia.edu
Dan Geer, OpenVision Technologies
geer@cam.ov.com
Phil Karn, Qualcomm
karn@qualcomm.com
Jim Kempf, Sun Microsystems
james.kempf@eng.sun.com
Jay Kistler, Digital Equipment Corporation
jjk@src.dec.com

Symposium Topics

We seek original and innovative papers about current developments in mobile and location-independent computing. We are especially interested in reports on practical experiences with mobile systems. The Mobile Computing Symposium will address a wide range of issues and ongoing developments, including, but not limited to:

- Applications for the mobile user
- Navigation and positioning (GPS, etc.)

- Security, especially in wireless environments or when away from home
- Caching and disconnected operation of applications and file systems
- Communications Protocols, including mobile TCP/IP
- Wireless communications (CDPD, CDMA, GSM, Ardis/RAM, cellular modem, etc.), and how they relate to and interact with operating systems and applications
- Portable and mobile computing equipment

Refereed Paper Submissions

Submission of an extended abstract of 1500–2500 words (9000–15000 bytes or 3–5 pages) is recommended. Shorter abstracts run a significant risk of rejection as there will be little on which the program committee can base an opinion. Extended abstracts should be sent to Jim Rees at the address below.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1-510-528-8649 or email to *mobile2authors@usenix.org*.

For administrative reasons (not blind reviewing), each submission should include a separate page or e-mail message giving the title of the paper, the names and affiliations of the authors, and the name of the author who will act as the contact person for the program committee. For the contact person, also include a daytime telephone number, postal address, e-mail address and FAX number if possible.

USENIX symposia, like most symposia and journals, require that papers not be submitted simultaneously to more than one conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non- disclosure agreement" forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

For More Program Information

For questions about refereed paper submissions and other program concerns, contact the Program Chair:

- Jim Rees
  CITI
  University of Michigan
  519 West William
  Ann Arbor, Michigan 48103 USA
- Internet: *Jim.Rees@umich.edu*
- Telephone: +1-313-763-4174
- Fax: +1-313-763-4434

Dates For Refereed Paper Submissions

- Extended abstracts due: January 2, 1995
- Notification to authors: January 23, 1995
- Camera-ready final papers due: March 6, 1995

For Registration Information

Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be available beginning in February 1995. If you wish to receive the registration materials, please contact USENIX at:

- USENIX Conference Office
  22672 Lambert Street
  Suite 613
  Lake Forest, CA 92630
  USA
- Telephone: +1-714-588-8649
- Fax: +1-714-588-9706
- Internet: *conference@usenix.org*

Workshop on Mobile Computing Systems and Applications

December 8-9, 1994

Santa Cruz, CA

Sponsored by the IEEE Computer Society TCOS in cooperation with ACM SIGOPS and USENIX Association

Call for Participation

General Chair
Darrell Long, *University of California, Santa Cruz*

Program Chair
M. Satyanarayanan, *Carnegie Mellon University*

Exhibits
Peter Honeyman, *University of Michigan*

Finance & Registration
Richard Golding, *Hewlett-Packard*

Publication
Luis-Felipe Cabrera, *IBM Almaden*

Program Committee
Dan Duchamp, *Columbia University*
Peter Honeyman, *University of Michigan*
Randy Katz, *UC Berkeley and ARPA*
Jay Kistler, *DEC SRC*
Krishan Sabnani, *AT&T Holmdel*
M. Satyanarayanan, *Carnegie Mellon University*
Amal Shaheen, *IBM Austin*
Marvin Theimer, *Xerox PARC*
Rich Wolff, *Bellcore*

A major challenge of this decade is the effective exploitation of two symbiotic technologies: portable computers and wireless networks. Harnessing these technologies will dramatically change the computing landscape. But realizing the full potential of the resulting mobile computing systems will require advances in many areas such as:

hardware • communications • scalability • power management • security • data access • user interfaces • location sensitivity

The goal of this workshop is to foster exchange of ideas in mobile computing among workers in the field. Attendance will be limited to about 60 participants, based on the position papers submitted. Submissions should be fewer than 5 pages in length and may expose a new problem, advocate a specific solution, or report on actual experience.

In addition, we will be hosting a small number of novel hardware and software exhibits relevant to mobile computing. The exhibits may be research prototypes or commercial products. Interested parties should submit technical descriptions of their exhibits.

Online copies of the position papers will be made available via anonymous FTP prior to the workshop. A printed proceedings will be published after the workshop, and mailed to participants.

A small number of graduate students will be granted a waiver of the registration fee. In return, these students will be required to take notes at the workshop and help put together the proceedings. Students who wish to be considered for the waiver must send in a brief description of their current research, and an explanation of how participation in the workshop is likely to help them.

Send 10 copies of position papers to:

M. Satyanarayanan
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
Email: *satya@cs.cmu.edu*
Phone: (412)-268-3743
Fax: (412)-681-5739

Send exhibit descriptions to:

Peter Honeyman
CITI University of Michigan
Ann Arbor, MI 48103-4943
Email: *honey@citi.umich.edu*
Phone: (313)-763-4413
Fax: (313)-763-4434

Important Dates

Submissions due	August 20, 1994
Acceptance Notification	October 1, 1994
Camera-ready copy due	November 15, 1994

GURU's 1994 Open Systems Conference & Exhibition: ROSE '94

November 3-4, 1994
Bucharest, Romania

First Announcement and Call for Papers

The second Conference & Exhibition specifically for Open Systems in Romania, ROSE '94, will be held on November 3-4, 1994, in Bucharest. The Event organiser is GURU – Romanian UNIX User Group, a member group of EurOpen – The European Forum for Open Systems.

The Conference's aim is to promote the knowledge and use of the information techniques based on Open Systems, by favouring the sharing of experiences and information between specialists with similar interests and the direct contact between users and suppliers of Open Systems products.

Two tracks are intended at the Conference: a technical track, in which specialists from Romania and abroad are invited to present papers on all topics related to Open Systems technology, and a business-oriented track, in which leading suppliers of software and hardware will present their strategy towards the Open Systems market; their products will be on show at the Exhibition organised together with the Conference.

The Event seeks to consolidate the success of the first ever Conference & Exhibition specifically for Open Systems in Romania, ROSE '93, held on September 30 and October 1, 1993, in Cluj-Napoca, which featured speakers from 8 countries and over 20 sponsor companies.

The theme of the ROSE '94 Conference is: "Open Systems, Technology for an Open World".

Submission of Papers

Topics for the Conference will cover the spectrum of recent research, development and experience using Open Systems technology. Papers are solicited on all aspects related to Open Systems, including:

- Architectures
- Operating systems
- Networking and Communications
- Software development tools and Applications
- Free software

Submissions should be in the form of extended abstracts or full papers (5-10 pages in length). The Conference language is English.

Please submit the papers, by post or e-mail, to Alexandru Rotaru, the programme chair, at:

GURU
P.O. BOX 63-42
Bucharest, Romania
<arot@guru.ro>

Accepted papers will be published in the Conference proceedings, which will be distributed to all the attendees. Presentations will usually be scheduled for 30 minutes.

Important Dates

September 3, 1994	Deadline for submission of extended abstracts or full papers
September 17, 1994	Notification of acceptance to authors
October 8, 1994	Deadline for receipt of the final papers

For further information, including participation conditions for companies, please contact:

Alexandru Rotaru (GURU's chairman)
GURU
P.O. BOX 63-42
Bucharest, Romania
Phone: (+40-1) 633 05 17 (home)
Fax: (+40-1) 312 27 45
E-mail: *<arot@guru.ro>*

Adrian Ivanov (vice-chairman)
ITC
109 Str. Republicii
3400 Cluj-Napoca, Romania
Phone: (+40-64) 11 60 60
Fax: (+40-64) 11 12 36
E-mail: *<aiva@cluj.iiruc.ro>*

Radu Togui (secretary)
GURU
P.O. BOX 63-42
Bucharest, Romania
Phone: (+40-1) 613 21 98
E-mail: *<radu@guru.ro>*

Daniel Dumitriu (board member)
P.O. BOX 61-33
Bucharest, Romania
E-mail: *<daniel@ddits10.guru.ro>*

Publications from John Wiley & Sons, Inc.

Member Number: _____

All USENIX members receive a 15% discount on the following Wiley publications

Introduction to Client/Server Systems
Paul Renaud
1-57774-X $34.95 ***member price: $29.71***
# of Copies: _____

Portable UNIX
Douglas Topham
1-57926-2 $14.95 ***member price: $12.71***
# of Copies: _____

The UNIX Command Reference Guide
Kaare Christian
1-85580-4 $32.95 ***member price: $28.01***
# of Copies: _____

C++ For Programmers
Leendert Ammeraal
1-93011-3 $32.95 ***member price: $28.01***
# of Copies: _____

Obfuscated C and Other Mysteries
Don Libes
1-57805-3 $39.95 ***member price: $33.96***
# of Copies: _____

Adventures in UNIX Network Applications Programming
Bill Rieken
1-52858-7 $34.95 ***member price: $29.71***
# of Copies: _____

UNIX, Self Teaching Guide
George Leach
1-57924-6 $19.95 ***member price: $16.96***
# of Copies: _____

UNIX Shell Programming, Second Edition
Lowell Jay Arhur
1-51821-2 $29.95 ***member price: $25.46***
# of Copies: _____

Object Oriented Programming withTurbo C++
Keith Weiskamp
1-52466-2 $24.95 ***member price: $21.21***
# of Copies: _____

Berkeley UNIX
A Simple & Comprehensive Guide
James Wilson
1-61582-X $37.95 ***member price: $32.26***
# of Copies: _____

U.S. orders only

☐ Payment enclosed, plus sales tax
☐ VISA ☐ Mastercard
☐ American Express
Card No. _____

Name _____
Firm _____
Address _____
City/State/Zip _____
Signature _____
(order invalid unless signed)

Please mail or fax orders to the following address:

John Wiley & Sons, Inc.
605 Third Avenue
New York, NY 10158
Attn: Karen Cooper

Phone: (212) 850-6789
FAX: (212) 850-6142

PTR Prentice Hall is pleased to recommend the following titles to USENIX members...

___**Object-Oriented Programming**, Peter Coad and Jill Nicola, 0-13-032616-X
(03261-5) *List:* $47.00 *Members:* $39.95

___**Internetworking with TCP/IP, Vol. III Client Server Programming and Applications for the BSD Socket Version,** Douglas E. Comer and David L. Stevens, 0-13-474222-2
(47422-1) *List:* $51.00 *Members:* $43.35

___**Internetworking with TCP/IP, Vol. III Client Server Programming and Applications for the AT&T TLI Version,** Douglas E. Comer and David L. Stevens, 0-13-474230-3
(47423-9) *List:* $50.00 *Members:* $42.50

___**The Internet Message: Closing the Book with Electronic Mail,** Marshall T. Rose, 0-13-092941-7
(09294-0) *List:* $46.00 *Members:* $39.10

___**Object-Oriented Databases: An Introduction,** Dimitris N. Chorafas and Heinrich Steinmann, 0-13-491804-5
(49180-3) *List:* $36.00 *Members:* $30.60

___**The Anatomy of Programming Languages** Alice E. Fischer and Frances S. Grodzinsky, 0-13-035155-5
(03515-4) *List:* $55.00 *Members:* $46.75

___**The Standard C Library,** PJ Plauger, 0-13-131509-9
(13150-8) *List:* $35.00 *Members:* $29.75

___**All About Administering the NIS+** Rick Ramsey, 0-13-068800-2
(06880-9) *List:* $35.00 *Members:* $29.75

___**The Simple Book: An Introduction to Internet Management** Marshall T. Rose, 0-13-177254-6
(17725-3) *List:* $55.00 *Members:* $46.75

___**Fiber Optics Networks,** Paul Green, Jr., 0-13-319492-2
(31949-1) *List:* $62.00 *Members:* $52.70

___**Networking Operations on UNIX SVR4,** Mike Padavano, 0-13-613555-2
(61355-4) *List:* $44.00 *Members:* $37.40

___**The User's Directory of Computer Networks,** Tracy L. LaQuey, 0-13-950262-9
(95026-1) *List:* $36.00 *Members:* $30.60

___**MIT Project Athena,** George A. Champine, 0-13-585324-9
(58532-3) *List:* $29.00 *Members:* $24.65

___**The Matrix,** John S. Quarterman, 0-13-565607-9
(56560-6) *List:* $50.00 *Members:* $42.50

___**Solaris Porting Guide,** Sunsoft ISV Engineering 0-13-030396-8
(03039-5) *List:* $49.00 *Members:* $41.65

___**Multiprocessor System Architectures,** Catanzaro 0-13-089137-1
(08913-6) *List:* $42.00 *Members:* $35.70

___**The HP-UX System Administrator's "How To" Book** Marty Poniatowski, 0-13-099821-4
(09982-0) *List:* $32.00 *Members:* $27.20

___**UNIX System V Performance Management,** Edited by Phyllis Eve Bregman and Sally A. Browning 0-13-016429-1
(01642-8) *List:* $29.95 *Members:* $25.45

___**SCO® UNIX® Operating System System Administrator's Guide,** Santa Cruz Operation, 0-13-012568-7
(01256-7) *List:* $38.00 *Members:* $32.30

MAIL, PHONE OR FAX YOUR ORDER!

Mention this code when calling: D582-91529-LF(2)

___ Check enclosed, made payable to: ***PTR Prentice Hall***

___ Please bill my: ___VISA ___ MasterCard ___AMEX

Card No. _____

Exp. Date _____ Signature _____

SHIP TO:

Name _____

Company _____

Address _____

_____

City _____ State _____ Zip _____

In USA:
PTR Div.—Prentice Hall Publishing
Box 11073
Des Moines, IA 50381-1073
515-284-6751 FAX 515-284-2607

In Canada:
Prentice Hall Canada
1870 Birchmount Rd.
Scarborough, ON M1P 2J7
416-293-3621 FAX 416-299-2540

In UK, Europe, Middle East & Africa:
Simon & Schuster Int'l
Campus 400, Marylands Ave.
Hemel Hempstead, Hertz HP2 7EZ
442-88 2163 FAX 442-88 2265

FOR MORE INFORMATION, OR QUANTITY ORDERS, PLEASE CALL 201-592-2657

N E T W O R K S

USENIX members receive a **15%** discount on the following MIT Press publications:

GLOBAL NETWORKS
Computers and International Communication
edited by Linda M. Harasim

Global Networks takes up the host of issues raised by the new networking technology that now links individuals, groups, and organizations in different countries and on different continents. The twenty-one contributions focus on the implementation, application, and impact of computer-mediated communication in a global context.

340 pp. $29.95 hardcover HARNH

THE NETWORK NATION
Human Communication via Computer Revised Edition
Starr Roxanne Hiltz and Murray Turoff

"*The Network Nation*... contained a fascinating vision. ...It is a place where thoughts are exchanged easily and democratically and intellect affords one more personal power than a pleasing appearance does. Minorities and women compete on equal terms with white males, and the elderly and handicapped are released from the confines of their infirmities to skim the electronic terrain as swiftly as anyone else." — Teresa Carpenter, *Village Voice*

580 pp. $24.95 paperback HILWP

THE EVOLUTION OF C++
Language Design in the Marketplace of Ideas
edited by Jim Waldo

This collection of articles traces the history of C++, from its infancy in the Santa Fe workshop, to its proliferation today as the most popular object-oriented language for microcomputers. Waldo notes in his postscript that in the process of evolving, the language has lost a clearly articulated, generally accepted design center, with no common agreement about what it should or should not do in the future.

279 pp. $24.95 paperback WALEP

TECHNOLOGY 2001
The Future of Computing and Communications
edited by Derek Leebaert

Researchers, executives, and strategic planners from inside the companies and laboratories that have shaped today's information age forecast the merging technologies that could well define the computing and communications environment that lies ahead.

392 pp. $14.95 paperback LEEEP

THE DIGITAL WORD
Text-Based Computing in the Humanities
edited by George P. Landow and Paul Delany

This book explores the larger realm of the knowledge infrastructure where texts are received, reconstructed, and sent over global networks.

Technical Communication and Information Systems series 384 pp. $39.95 hardcover LANDH

SOCIOMEDIA
Multimedia, Hypermedia, and the Social Construction of Knowledge
edited by Edward Barrett

Sociomedia continues the assessment of hypertext and hypermedia systems begun in *Text, ConText, and HyperText* and *The Society of Text*. It examines the use of integrated multimedia to support social or collaborative research, learning, and instruction in the university, one of the best environments for developing and analyzing the effects of computing technologies on our understanding of complex sets of information.

Technical Communications and Information Series 360 pp. $50.00 hardcover BARRH

CONNECTIONS
New Ways of Working in the Networked Organization
Lee Sproull and Sara Kiesler

"...Sproull and Kiesler raise crucial questions about our technical and particularly our human strategies as a producing society."
— Howard Webber, *Sloan Management Review*

228 pp. $21.95 paperback SPRCP

TECHNOBABBLE
John A. Barry

"A serious study of the language of the new technocracy."
— William Safire, *The New York Times Magazine*

288 pp. $12.50 paperback BARCP

Please send me these titles _____

☐ Payment enclosed ☐ Purchase Order Attached Charge to my: ☐ Master Card ☐ VISA

Card # _____ exp. _____ Signature _____

$ _____ Total for book(s)
$ 3.00 Postage for North American addresses
$ _____ Canadian customers add 7% GST
$ _____ Total for book(s) & postage

Name _____
Address _____
City _____ State _____ Zip _____
Phone _____ FAX _____

Make checks payable and send order to:

THE MIT PRESS
55 Hayward Street, Cambridge, MA 02142-1399 USA

To order by phone, call (617) 625-8569 or (800) 356-0343. E-mail order # mitpress-orders@mit.edu. The operator will need this code: **UNIX1.**

McGraw–Hill Books
20% Discount for USENIX Members

Qty

ection Its Architecture and Protocols, Revised Edition..... ☐

Bijendra N. Jain & Ashok K. Agrawala 0-07-032385-2 hardcover
retail: $50.00 member price: $40.00

A Students Guide to UNIX..... ☐

Harley Hahn 0-07-025511-3 paperback
retail: $25.75 member price: $20.60

Data Network Design Packet Switching, Frame Relay, 802.6/DQDB..... ☐

Darren L. Spohn 0-07-060360-X hardcover
retail: $59.00 member price: $47.20

Migrating to Open Systems: Taming the Tiger..... ☐

Daniel R. Perley, 0-07-707778-4 hardcover
retail: $30.00 member price: 24.00

I am a member of the USENIX Association. Please send me the books I have indicated at the member special rate. I have added $3.00 postage and handling for the first book ordered, $1.00 for each additional book, plus my local sales tax.

Total price of books _____
Sales tax _____
Postage & Handling _____
Total enclosed _____

USENIX Membership #_____

____ Check or money order is enclosed - payable to McGraw-Hill, Inc.

____ Charge my ____ Visa ____ Mastercard ____ Discover ____ Amex

Account #_____

Expiration Date_____

Signature_____

Shipping Information (7/93 - 03US001)

Name _____

Address _____

_____

Daytime Phone # _____

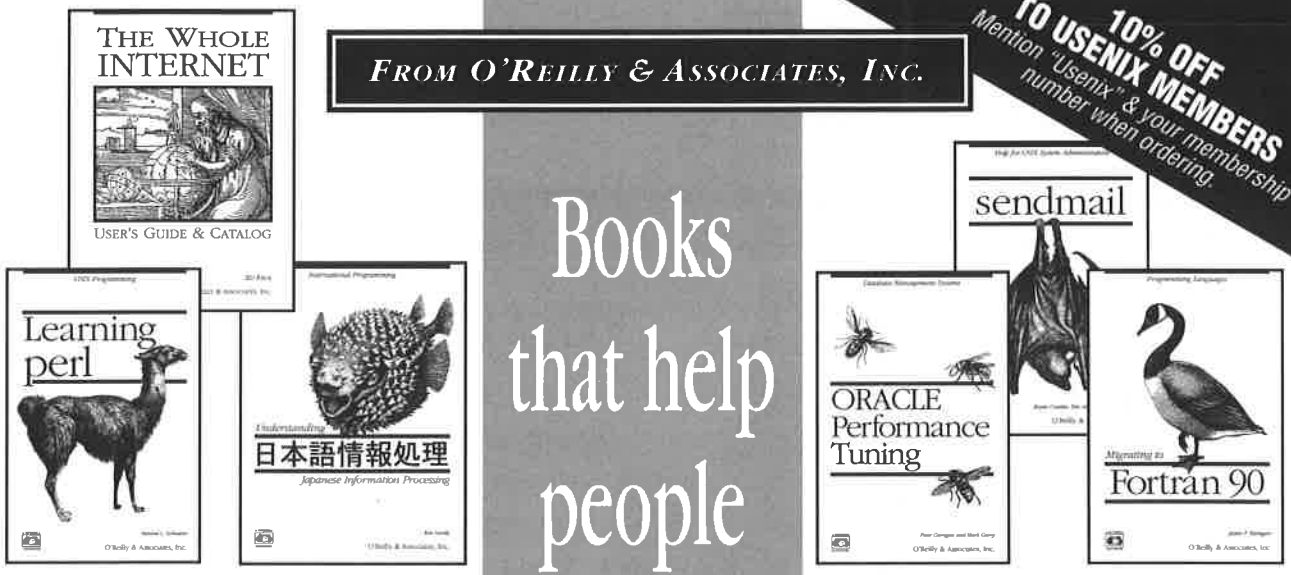
Send orders to:

McGraw-Hill, Inc.
Attn: Rosa Perez
11 West 19th Street - 4th Floor
New York, New York 10011

If I am not completely satisfied, I will return the book(s) within 15 days or a full refund or credit.
Satisfaction unconditionally guaranteed. Prices subject to change without notice. We can only accept orders within the continental USA.

FROM O'REILLY & ASSOCIATES, INC.

10% OFF TO USENIX MEMBERS
Mention "Usenix" & your membership number when ordering.



Books that help people get more out of computers

The Whole Internet User's Guide & Catalog

By Ed Krol, 1st Edition September 1992
400 pages, ISBN 1-56592-025-2

A comprehensive—and best-selling—introduction to the Internet, the international network that includes virtually every major computer site in the world.
This book is a comprehensive introduction to what's available and how to find it. In addition to electronic mail, file transfer, remote login, and network news, *The Whole Internet* pays special attention to some new tools for helping you find information. Whether you're a researcher, a student, or just someone who likes electronic mail, this book will help you to explore what's possible.

Learning perl

By Randal L. Schwartz
1st Edition November 1993
274 pages, ISBN 1-56592-042-2

Perl is rapidly becoming the "universal scripting language." Combining capabilities of the UNIX shell, the C programming language, *sed*, *awk*, and various other utilities, it has proved its use for tasks ranging from system administration to text processing and distributed computing. *Learning perl* is a step-by-step, hands-on tutorial designed to get you writing useful perl scripts as quickly as possible. In addition to countless code examples, there are numerous programming exercises, with full answers. For a comprehensive and detailed guide to programming with Perl, read O'Reilly's companion book *Programming perl*.

Understanding Japanese Information Processing

By Ken Lunde
1st Edition September 1993
470 pages, ISBN 1-56592-043-0

Understanding Japanese Information Processing provides detailed information on all aspects of handling Japanese text on computer systems. It covers everything from the origins of modern-day Japanese to the latest information on specific emerging computer encoding standards.

sendmail

By Bryan Costales, with Eric Allman & Neil Rickert
1st Edition November 1993
830 pages, ISBN 1-56592-056-2

Although *sendmail* is used on almost every UNIX system, it's one of the last great uncharted territories—and most difficult utilities to learn—in UNIX system administration. This book provides a complete *sendmail* tutorial, plus extensive reference material. It covers the BSD, UIUC IDA, and V8 versions of *sendmail*. Part One of the book is a tutorial on understanding *sendmail*; Part Two covers practical issues in *sendmail* administration; Part Three is a comprehensive reference section; and Part Four consists of appendices and a bibliography.

ORACLE Performance Tuning

By Peter Corrigan & Mark Gurry
1st Edition September 1993
642 pages, ISBN 1-56592-048-1

The ORACLE relational database management system is the most popular database system in use today. This book shows you the many things you can do to dramatically increase the performance of your existing ORACLE system, whether you are running RDBMS Version 6 or Version 7. You may find that this book can save you the cost of a new machine; at the very least, it will save you a lot of headaches.

Migrating to Fortran 90

By James F. Kerrigan
1st Edition November 1993 (est.)
315 pages (est.), ISBN 1-56592-049-X

This book is a practical guide to Fortran 90 for the current Fortran programmer. It provides a complete overview of the new features that Fortran 90 has brought to the Fortran standard, with examples and suggestions for use. Topics include array sections, modules, allocatable arrays and pointers, file handling, and numeric precision.

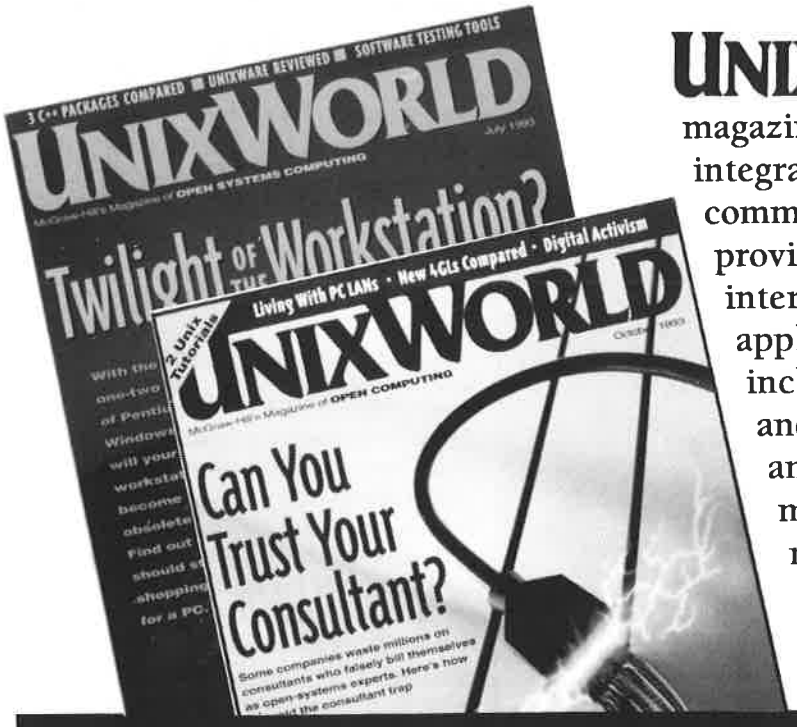
O'Reilly & Associates, Inc.

103A Morris Street, Sebastopol, California 95472 • 800/998-9938 • 707/829-0515 • FAX 707/829-0104 • order@ora.com

Special Offer

USENIX Association Members

THE WORLDWIDE COMPUTING AUTHORITY



UNIXWORLD is a monthly magazine written for people who integrate, resell, manage, and program commercial computer systems to provide solutions based on interoperable networks and applications. Topics for features include the effective use of UNIX and other open systems to downsize and distribute applications. The magazine also provides a useful mix of product reviews, case studies, industry profiles and analysis, and technical tutorials.

THE BENEFITS OF UNIXWORLD:

Programming Tips - helps the reader become educated in UNIX and OPEN SYSTEMS computing. Provides real-world application information the reader can use everyday.

Unbiased Hardware and Software Reviews - helps the reader recommend and justify purchasing decisions.

Clear Editorial - so both beginners and experts can get full benefit from the publication. Product Trends and Technical Advances - to aid professionals in developing company-wide systems strategies.

Profiles on U.S. and International Companies - provides the reader with a truly global perspective of the marketplace and the players. Helps vendors assess and monitor the competition.

Special Rates:

DOMESTIC		CANADIAN		OTHER INTERNATIONAL (Air Delivery)	
1 Yr.	$15.00	1 Yr.	$21.00	1 Yr.	$ 48.00
2 Yrs.	$24.00	2 Yrs.	$36.00	2 Yrs.	$ 90.00
3 Yrs.	$30.00	3 Yrs.	$48.00	3 Yrs.	$130.00

Be wise subscribe today!

To place your order please call:

Kelly Kendall at 1-800-327-6850 or fax your order to 609/426-7055

McGraw-Hill, Inc.• Dept. 418 • Hightstown, NJ 08520

ORDER FORM

CONFERENCE & WORKSHOP PROCEEDINGS

USENIX

Qty	Proceedings		Member Price	Non-Member Price	Subtotal	Overseas Postage	Total
WINTER/SUMMER CONFERENCES							
____	Boston	Summer '94	25	33	$_____	$20	$_____
____	San Francisco	Winter '94	30	39	$_____	$20	$_____
____	Cincinnati	Summer '93	25	33	$_____	18	$_____
____	San Diego	Winter '93	33	40	$_____	25	$_____
____	San Antonio	Summer '92	23	30	$_____	14	$_____
____	San Francisco	Winter '92	30	39	$_____	22	$_____
____	Nashville	Summer '91	32	38	$_____	22	$_____
____	Dallas	Winter '91	28	34	$_____	18	$_____
____	Anaheim	Summer '90	22	22	$_____	15	$_____
____	Washington, DC	Winter '90	25	25	$_____	15	$_____
____	Baltimore	Summer '89	20	20	$_____	15	$_____
____	San Diego	Winter '89	30	30	$_____	20	$_____
____	San Francisco	Summer '88	29	29	$_____	20	$_____
____	Dallas	Winter '88	26	26	$_____	15	$_____
____	Phoenix	Summer '87	35	35	$_____	20	$_____
____	Washington, DC	Winter '87	10	10	$_____	15	$_____
____	Atlanta	Summer '86	37	37	$_____	20	$_____
____	Denver	Winter '86	25	25	$_____	15	$_____
____	Portland	Summer '85	45	45	$_____	25	$_____
____	Dallas	Winter '85	15	15	$_____	10	$_____
____	Salt Lake City	Summer '84	29	29	$_____	20	$_____
____	Washington, DC	Winter '84	25	25	$_____	15	$_____
____	Toronto	Summer '83	32	32	$_____	20	$_____
____	San Diego	Winter '83	28	28	$_____	15	$_____
LARGE INSTALLATION SYSTEMS ADMINISTRATION							
____	SANS III	April '94	15	20	$_____	9	$_____
____	LISA VII	Nov. '93	25	33	$_____	12	$_____
____	LISA VI	Oct. '92	23	30	$_____	12	$_____
____	LISA V	Sept. '91	20	23	$_____	11	$_____
____	LISA IV	Oct. '90	15	18	$_____	8	$_____
____	LISA III	Sept. '89	13	13	$_____	9	$_____
____	LISA II	Nov. '88	8	8	$_____	5	$_____
____	LISA I	April '87	4	4	$_____	5	$_____
C++							
____	C++ Conference	April '94	24	28	$_____	20	$_____
____	C++ Conference	Aug. '92	30	39	$_____	20	$_____
____	C++ Conference	April '91	22	26	$_____	11	$_____
____	C++ Conference	April '90	28	28	$_____	18	$_____
____	C++ Conference	Oct. '88	30	30	$_____	20	$_____
____	C++ Workshop	Nov. '87	30	30	$_____	20	$_____
SECURITY							
____	UNIX Security IV	Oct. '93	15	20	$_____	9	$_____
____	UNIX Security III	Sept. '92	30	39	$_____	11	$_____
____	UNIX Security II	Aug. '90	13	16	$_____	8	$_____
____	UNIX Security	Aug. '88	7	7	$_____	5	$_____
MACH							
____	Mach Symposium III	April '93	30	39	$_____	18	$_____
____	Mach Symposium	Nov. '91	24	28	$_____	14	$_____
____	Mach Workshop	Oct. '90	17	20	$_____	9	$_____

Qty	Proceedings		Member Price	Non-Member Price	Subtotal	Overseas Postage	Total
	DISTRIBUTED & MULTIPROCESSOR SYSTEMS (SEDMS)						
___	SEDMS IV	Sept. '93	24	32	$_____	14	$_____
___	SEDMS III	Mar. '92	30	36	$_____	20	$_____
___	SEDMS II	Mar. '91	30	36	$_____	20	$_____
___	SEDMS	Oct. '89	30	30	$_____	20	$_____
	MICROKERNELS & OTHER KERNEL ARCH.						
___	Microkernels & Other Kernel... II	Sept. '93	15	20	$_____	9	$_____
___	Microkernels & Other Kernel... I	Apr. '92	30	39	$_____	20	$_____
	GRAPHICS						
___	Graphics Workshop V	Nov. '89	18	18	$_____	10	$_____
___	Graphics IV	Oct. '87	10	10	$_____	10	$_____
___	Graphics III	Nov. '86	10	10	$_____	5	$_____
___	Graphics II	Dec. '85	7	7	$_____	5	$_____
	OTHER WORKSHOPS/SYMPOSIA						
___	UNIX Applications Development	April '94	15	20	$_____	9	$_____
___	Mobile Computing	Aug. '93	15	20	$_____	8	$_____
___	File Systems	May '92	15	20	$_____	9	$_____
___	UNIX Transaction Processing	May '89	12	12	$_____	8	$_____
___	Software Management	Apr. '89	20	20	$_____	15	$_____
___	UNIX & Supercomputers	Sept. '88	20	20	$_____	10	$_____

Discounts are available for bulk orders. Please inquire.

Total price of Proceedings _____**
Calif. residents add sales tax _____
Total overseas postage _____
Total enclosed _____

****If you are paying member price, please include member's name and/or membership number**_____

PAYMENT OPTIONS*

___ Check enclosed - payable to USENIX Association

___ Charge my: ___ VISA ___ MC Account # _____ Exp.Date _____

___ Purchase order enclosed Signature _____

* Outside the USA? Please make your payment in US currency by one of the following:
- Check - issued by a local branch of a US Bank
- Charge (VISA, MasterCard, or foreign equivalent)
- International postal money order

Shipping Information

Please allow 2–3 weeks for delivery. Overseas orders are shipped via air printed matter.

Please mail or fax this order form with payment to:

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
FAX 510/548-5738

Ship to: _____

• If you are not a member and wish to receive our membership information packet, please check this box. ☐

USENIX

SUBSCRIPTION ORDER FORM

PROCEEDINGS

Please enter my one-year subscription* to the 1994 USENIX Conferences and Symposia Proceedings, which includes:

San Francisco Winter Technical Conference - January 1994

C++ Conference - April 1994

Applications Development Symposium - April 1994

Boston Summer Technical Conference - June 1994

Symposium on High-Speed Networking - August 1994

Systems Administration (LISA VIII) Conference - September 1994

Symposium on Very High Level Languages (VHLL) - October 1994

Symposium on Operating Systems Design and Implementation (OSDI) - November 1994

*NOTE:
Corporate, Educational and Supporting Members of USENIX automatically receive a subscription to all proceedings published during their term of membership, which may overlap with this offer.

COST: USENIX members: $190.00
Non-members: $235.00
Outside U.S.A. and Canada?
Add $30.00 for surface postage.

Subscription Cost _____ **
Calif. Sales Tax _____
Overseas Postage _____
TOTAL ENCLOSED _____

** *If you are paying member price, please include member's name and/or membership number* _____

PAYMENT OPTIONS

☐ Check enclosed payable to USENIX Association. ☐ Purchase order enclosed.
☐ Please charge my: ☐ Visa ☐ MasterCard

Account # _____ Exp. Date _____

Signature _____

Outside the U.S.A.? Please make your payment in U.S. currency by one of the following:
* Check - issued by a local branch of a U.S. Bank
* Charge (Visa, MasterCard, or foreign equivalent)
* International postal money order

1/94

Name _____

Address _____

City _____ State/Country _____ Zip/Postal Code _____

Please mail this order form to: USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
510/528-8649
FAX 510/548-5738
office@usenix.org

This offer is good through December 31, 1994

The proceedings are shipped the week following the event.

The Association will support local user groups by doing a mailing to assist in the formation of a new group and publishing information on local groups in *;login:*. At least one member of the group must be a current member of the Association. Send additions and corrections to: <*login@usenix.org*>.

California

Fresno:

The Central California UNIX Users Group consists of a uucp-based electronic mailing list to which members may post questions or information. For connection information:

- Educational and governmental institutions:
  Brent Auernheimer
  (209) 278-2573,
  <*brent@CSUFresno.edu*> or
  <*csufres!brent*>
- Commercial institutions or individuals:
- Gordon Crumal
  (209) 251-2648
  <*csufres!gordon*>

Orange County

Meets the 2nd Monday of each month

- UNIX Users Association of Southern California
  Paul Muldoon
  (714) 556-1220 ext. 137
  New Horizons Computer Learning Center
  1231 E. Dyer Rd., Suite 140
  Santa Ana, CA 92705

Colorado

Boulder

Meets monthly at different sites. For meeting schedule, send email to <*fruug-info@fruug.org*>.

- Front Range UNIX Users Group
  Lone Eagle Systems Inc.
  636 Arapahoe #10
  Boulder, CO 80302
  Steve Gaede
  (303) 444-9114
  <*gaede@fruug.org*>

Washington, D.C.

Meets 1st Tuesday of each month.

- Washington Area UNIX Users Group
  9811 Mallard Drive
  Laurel, MD 20708
  Alan Fedder
  (301) 953-3626

Florida

Coral Springs:

- S. Shaw McQuinn
  (305) 344-8686
  8557 W. Sample Road
  Coral Springs, FL 33065

Melbourne:

Meets the 3rd Monday of every month.

- Space Coast UNIX User's Group
  Steve Lindsey
  (407) 242-4766
  <*lindsey@vnet.ibm.com*>

Orlando:

Meets the 3rd Thursday of each month.

- Central Florida UNIX Users Group Mikel Manitius
  (407) 444-8448
  <*mikel@aaa.com*>

Western:

Meets 1st Thursday of each month.

- Florida West Coast UNIX Users Group
  Richard Martino (813) 536-1776
  Tony Becker (813) 799-1836
  <*mcrsys!tony*>
  Ed Gallizzi, Ph.D. (813) 864-8272
  <*e.gallizzi@compmail.com*>
  Jay Ts (813) 979-9169
  <*uunet!pdn!tscs!metran!jan*>
  Dave Lewis (407)242-4372
  <*dhl@ccd.harris.com*>

Georgia

Atlanta:

Meets on the 1st Monday of each month in White Hall, Emory University.

- Atlanta UNIX Users Group
  P.O. Box 12241
  Atlanta, GA 30355-2241
  Mark Landry (404) 365-8108

Kansas or Missouri

Meets on 2nd Monday of each month.

- Kansas City UNIX Users Group (KUUG)
  813B Street
  Blue Springs, MO 64015
  (816) 235-5212
  <*mlg@cstp.umkc.edu*>

Michigan

Detroit/Ann Arbor

Meets on the 2nd Thursday of each month in Ann Arbor.

- Southeastern Michigan Sun Local Users Group and Nameless UNIX Users Group
  Steve Simmons office:
  (313)769-4086
  home: (313) 426-8981
  <*scs@lokkur.dexter.mi.us*>

Minnesota

Minneapolis/St. Paul:

Meets the 1st Wednesday of each month.

- UNIX Users of Minnesota
  17130 Jordan Court
  Lakeville, MN 55044
  Robert A. Monio
  (612) 220-2427
  <*pnessutt@dmshq.mn.org*>

Missouri

St. Louis:

- St. Louis UNIX Users Group P.O. Box 2182 St. Louis, MO 63158
  Terry Linhardt
  (314) 772-4762
  <*uunet!jgaltstl!terry*>

Nebraska

Omaha: Meets monthly.

- /usr/group/nebraska
  P.O. Box 31012
  Omaha, NE 68132
  Phillip Allendorfer
  (402) 423-1400

New England

Northern:
Meets monthly at different sites.

- Peter Schmitt (603) 646-2085
  Kiewit Computation Center
  Dartmouth College
  Hanover, NH 03755
  <peter.schmitt@dartmouth.edu>

New Jersey

Princeton:
Meets monthly.

- Princeton UNIX Users Group
  Mercer County Community College
  1200 Old Trenton Road
  Trenton, NJ 08690
  Peter J. Holsberg (609) 586-4800
  <mccc!pjh>

New Mexico

Albuquerque:
ASIGUNIX meets every 3rd Wednesday of each month.

- Phil Hortz 505/275-0466.

New York

New York City:
Meets every other month in Manhattan.

- Unigroup of New York City
  G.P.O.
  Box 1931
  New York, NY 10116
  <unigroup@murphy.com>
  Bob Young (212) 490-8470

Oklahoma

Tulsa:
Meets 2nd Wednesday of each month.

- Yulsa UNIX Users Group, $USR
  Stan Mason (918) 560-5329
  <tulsix!smason@drd.com>
  Mark Lawrence (918) 743-3013
  <mark@drd.com>

Texas

Austin:
Meets 3rd Thursday of each month.

- Capital Area Central Texas UNIX Society (CACTUS)
  P.O. Box 9786
  Austin, TX 78766-9786
  Tom Painter (512) 258-7321
  <president@cactus.org>

Dallas/Fort Worth:
Meets the 1st Thursday of each month.

- Dallas/Fort Worth UNIX Users Group
  P.O. Box 867405
  Plano, TX 75086
  Evan Brown (214) 519-3577
  <evbrown@dsccc.com>

Houston:
Meets 3rd Tuesday of each month.

- Houston UNIX Users Group
  (Hounix) answering machine
  (713) 684-6590
  Bob Marcum, President
  (713) 626-4100
  Chuck Bentley, Vice-president
  (713) 789-8928
  <chuckb@hounix.uucp>

Washington

Seattle:
Meets monthly.

- Seattle UNIX Group Membership Info.
  Bill Campbell (206) 947-5591
  6641 East Mercer
  Mercer Island, WA 98040-0820
  <bill@celestial.com>

Canada

Manitoba:
Meets 2nd Tuesday of each month.

- Manitoba UNIX User Group
  (MUUG) P.O. Box 130
  St. Boniface Winnipeg,
  MB R2H 3B4
  Bary Finch, President
  (204) 934-2723
  <info@muug.mb.ca>

Ottawa:

- The Ottawa Carleton UNIX Users Group
  D.J. Blackwood
  (613)957-9305
  <dave@revcan.rct.ca>

Toronto:

- 143 Baronwood Court
  Brampton, Ontario
  Canada L6V 3H8
  Evan Leibovitch
  (416) 452-0504
  <evan@telly.on.ca>

System Administration Groups

Back Bay LISA (BBLISA)

New England forum covering all aspects of system and network administration, for large and small installations. Meets monthly, at MIT in Cambridge, MA.
For information, contact:

- J. R. Oldroyd (617)227-563
  <jr@opal.com>
- Mailing list subscription:
  <requests:bblisaquest@cs.umb.edu>
- Mailing list postings:
  <bblisa@cs.umb.edu>
- For current calendar of events:
  finger *<bblisa@cs.umb.edu>*

Bay LISA

Meets 3rd Thursday of each month in Mountain View, CA For more information, please contact:
<baylisa-info@baylisa.org> or

- Bryan McDonald,
  BayLISA President
  <bigmac@baylisa.org>
  P.O. Box 64369
  Sunnyvale CA, 94088-4369
  (415) 859-3246

$GROUPNAME (New Jersey)

$GROUPNAME is an organization in New Jersey formed to facilitate information exchange pertaining to the field of UNIX system administration. For more information, send email to:
Majordomo@Warren. MENTORG.COM or
Tom Limoncelli
<tom_limoncelli@warren.mentorg.com>

New York Systems Administrators (NYSA)

Meets 2nd Monday of each month.

- *<nysa-request@esm.com>*
  914/472-3635

North Carolina System Administrators Group

The North Carolina System Administrators Group meets on the 2nd Monday each month around the Research Triangle Park area.

- Amy Kreiling (919) 962-1843
  <kreiling@cs.unc.edu>
- William E. Howell (919) 962-1717
  <howell@cs.unc.edu>

ACM:
Association for Computing Machinery

AFUU:
Association of French UNIX Users

AUUG:
Australian UNIX Users Group

COOTS:
Conference on Object-Oriented Technology

DECUS:
Digital Equipment Computer Users Society

EurOpen:
European Forum for Open Systems

FedUNIX:
Council of Advanced Computing Systems Technologists in Government

GURU:
Roumanian UNIX User Group

GUUG:
German UNIX Users Groups

IEEE: Institute of Electrical and Electronics Engineers

IETF:
Internet Engineering Task Force

INET:
Internet Society

Interex:
International Association of Hewlett-Packard Computer Users

JUS:
Japan UNIX Society

LISA:
USENIX Systems Administration Conference

NOSSDAV:
Network and OS Support for Digital Audio and Video

OOPSLA:
Object-oriented Programming Systems, Languages and Applications

OSDI:
Symposium on Operating Systems Design & Implementation

SAGE:
System Administrators Guild

SANS:
System Administration, Networking & Security

SUG:
Sun User Group

SUUG:
Soviet UNIX Users Group

UKUUG:
United Kingdom UNIX Systems Users Group

UniForum:
International Association of UNIX and Open Systems Professionals

CALENDAR OF EVENTS

This is a combined calendar of planned conferences, symposia, and standards meetings related to the UNIX operating system. If you have a UNIX-related event that you wish to publicize, please contact <*login@usenix.org*>. Please provide your information in the same format as above.

* = events sponsored by the USENIX Association.

1994

August

1- 2* High-Speed Networking, Berkeley, CA

September

6 - 9 AUUG, Melbourne, Australia
12-14 NetWorld+INTEROP 94, Atlanta, GA
18 -22 Interex 94, Denver, CO
20 -22 GUUG, Wiesbaden, Germany
19 -23* LISA VIII, San Diego, CA

October

12-13 AFUU, Grenoble, France
17-21 IEEE 1003, Seattle, WA
23-27 ACM OOPSLA, Portland, OR
26-28* Very High Level Languages, Santa Fe, NM

November

3- 4 GURU, Bucharest, Romania
14-18* OSDI, Monterey, CA
14-18 SUG Technical Workshop, Austin, TX

December

8 - 9* IEEE Mobile Computing Systems & Applications, Santa Cruz, CA

1995

January

9-13 IEEE 1003
16-20* USENIX, New Orleans, LA

March

13-17 UniForum, Dallas, TX

April

SANS IV, Washington, DC
10-11* 2nd Mobile and Location–Independent Computing, Ann Arbor, MI
10-14 IEEE 1003

May

6-11 DECUS, Washington, DC

June

26 - 29* COOTS, Monterey, CA

July

10-14 IEEE 1003

August

6-11 Siggraph, Los Angeles, CA
13-17 Interex 95, Toronto, Canada

September

18-22* LISA IX, Monterey, CA

October

9-13 IEEE 1003

November

2- 8 DECUS, San Francisco, CA

1996

January

22-26* USENIX, San Diego, CA

February

14-16 UniForum, San Francisco, CA

August

4 - 8 Interex 96, San Diego, CA

November

16-22 DECUS, Anaheim, CA

When your business depends on the Internet, connect to the Internet service provider with the most experience in the business.

AlterNet®

A service of UUNET® Technologies, Inc.

A Full Spectrum of Services:

- Complete range of dedicated connections, all with no usage charges and unlimited e-mail, news, ftp, telnet, gopher, WWW
- Low-priced dial-up services with full IP access
- Nationwide points-of-presence
- Equipment configuration, domain name and IP address registration at no additional charge
- Free access to UUNET's extensive software archives
- On-site consulting services

Technical Excellence:

- Technical assistance from the Internet's top engineers, both during and after service installation
- Fully-staffed Network Operations Center, 24 hours a day/365 days a year

External Connectivity:

- Direct connections to more of the Internet's component networks worldwide than any other service provider
- Founding member of the CIX and MAE-East

Network Superiority:

- The Internet's first 45 Mbps ATM backbone
- Customer access up to 10 Mbps
- More redundant connections within our network than any other Internet service provider, minimizing customer exposure to network outages

For more information, contact an AlterNet representative.

Voice: +1 800 258 4037
E-mail: alternet-info@alter.net

Serving your Internet needs since 1987

SECOND CLASS POSTAGE
PAID
AT BERKELEY, CALIFORNIA
AND ADDITIONAL OFFICES

USENIX Association
2560 Ninth Street
Suite 215
Berkeley, CA 94710

POSTMASTER:
SEND ADDRESS CHANGES
TO ***;login:***
USENIX ASSOCIATION
2560 NINTH STREET
SUITE 215
BERKELEY, CA 94710